



Signaux avec Matlab

&

Introduction aux analyses spectrales

Durée: 8 périodes (2 séances de laboratoire)
Travail **individuel**

1. Objectifs

L'exercice a pour objectifs la compréhension et la mise en œuvre des concepts suivants :

- Utilisation de Matlab en général et de fonctions spécifiques en particulier.
- Construction et affichage de signaux élémentaires.
- Appliquer et interpréter une analyse fréquentielle sur un signal.

2. Environnement

La donnée et les fichiers additionnels éventuels sont disponibles sous :

`T:\EIFR\don_prof\Oberson\SystèmesElectronique\Labo\lab1\`

Le travail à réaliser sera fait entièrement dans l'environnement de Matlab à l'aide d'un ou de plusieurs script(s).

3. Travail à réaliser

Créez un nouveau script Matlab et sauvez-le en le nommant de la manière suivante:

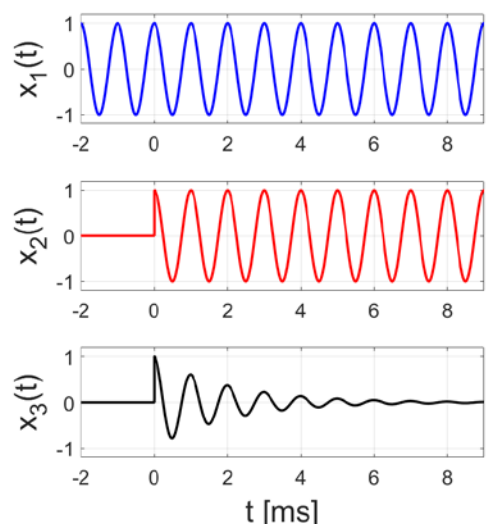
`lab1_<votre nom>.m`

3.1 Signal entré point à point

Afin de se familiariser avec la construction des signaux et leur affichage dans Matlab. Générez les signaux définis ci-dessous et développez le script pour que l'affichage corresponde à celui-ci-contre.

$$\left. \begin{aligned} x_1(t) &= \cos(2\pi f \cdot t) \\ x_2(t) &= \cos(2\pi f \cdot t) \cdot u(t) \\ x_3(t) &= e^{-\frac{t}{\tau}} \cos(2\pi f \cdot t) \cdot u(t) \end{aligned} \right\} \text{ où: } \begin{aligned} f &= 1\text{kHz} \\ \tau &= 1\text{ms} \end{aligned}$$

Pour cela, intéressez-vous notamment aux fonctions `plot()`, `subplot()`, `xlim()`, `xlabel()`, `heaviside()` de Matlab.



3.2 Série

Faites une nouvelle section dans votre script Matlab avec %% et générez les signaux suivant:

$$x_1(t) = \sin(2\pi ft) + \frac{1}{3}\sin(2\pi 3ft) + \frac{1}{5}\sin(2\pi 5ft) + \frac{1}{7}\sin(2\pi 7ft) \dots$$

$$x_2(t) = \cos(2\pi ft) + \frac{1}{3^2}\cos(2\pi 3ft) + \frac{1}{5^2}\cos(2\pi 5ft) + \frac{1}{7^2}\cos(2\pi 7ft) \dots$$

où $f = 2\text{Hz}$ et $-1[s] < t < 1[s]$ avec une résolution de $10\mu\text{s}$.

Affichez les deux signaux sur le même graphique. Quels signaux obtient-on lorsque l'on augmente le nombre de termes des sommes?

3.3 Représentation fréquentielle

Comme tous les signaux périodiques peuvent être décomposés en somme de sinus et/ou cosinus comme pour les exemples de l'exercice précédent, il est intéressant de savoir comment déterminer cette somme. Cela se fait par la série de Fourier.

Cet exercice permet d'avoir un exemple du calcul de cette série de Fourier d'un signal numérique sur Matlab qui pourra être utilisé comme base pour des analyses fréquentielles dans des projets futurs.

En vous appuyant sur la documentation en annexe, calculez dans Matlab la FFT des signaux $x_1(t)$ et $x_2(t)$ avec la fondamentale et 3 harmoniques sans fenêtre de pondération et affichez le résultat graphiquement.

Déterminez depuis le résultat des calculs des FFT, les amplitudes et les phases des composantes fréquentielles $x_1(t)$ et $x_2(t)$.

3.4 Analyse fréquentielle

A l'aide de la fonction de Matlab `load <filename>`, chargez les fichiers `piano.mat` et `guitare.mat` disponible sous *don_prof*. Utilisez alors la fonction `sound()` de Matlab pour écouter la note de piano et la note de guitare enregistrées. Puis, en faisant une analyse fréquentielle par la FFT comme précédemment, déterminez la fréquence des notes enregistrées dans ces deux fichiers.

Ces notes sont-elles différentes? Le son entendu est-il différent? Qu'en concluez-vous.

3.5 Optionnel

- 1) Développez une fonction `serie()` pour générer automatiquement le signal $x_1(t)$ ou le signal $x_2(t)$ du point 3.2 avec un nombre configurable de composantes.
- 2) Recréez le signal $x_1(t)$ avec la fondamentale et 3 composantes, mais cette fois pour $0[s] < t < 2.25[s]$ avec une résolution de $10\mu\text{s}$. Refaites sur ce signal votre analyse fréquentielle du point 3.3. Que constatez-vous?

Afin de corriger cette analyse fréquentielle, appliquez une fenêtre de pondération flat-top en vous inspirant de la documentation annexe "Introduction pratique à l'analyse de Fourier".

- 3) En utilisant la FFT des sons de piano et/ou de guitare, décalez le spectre afin que la fondamentale soit à une fréquence de 880Hz. Pour cela ajoutez des 0 au centre du spectre en utilisant la fonction `zeros()` de Matlab et reconstruisez un vecteur `fftson2`:

```
yfft=[xfft(round(440/(Fs/N)+1):N/2) zeros(1,round(2*440/(Fs/N))-1)
      xfft(N/2:N-round(440/(Fs/N)+1))];
```

Puis, faites la iFFT en utilisant la fonctions `ifftshift()`

```
son=real(ifft(ifftshift(N*yfft)));
```

Ecoutez le son obtenu à l'aide de `sound()` et affichez le avec `plot()`. Expliquez ce qui a été modifié sur le spectre. Quels sont les caractéristiques et la qualité du son de synthèse généré?

4. Rendu

Le rendu de cette séance d'exercices est composé du(des) script(s) Matlab ainsi que d'un rapport avec les résultats du travail effectué et les analyses de ces résultats. Le(s) script(s) et le rapport en format pdf doivent être envoyés par mail à Daniel Oberson daniel.oberson@hefr.ch au plus tard une semaine après la deuxième séance de TP.

5. Références

Aide en ligne de Matlab : <https://ch.mathworks.com/matlabcentral/>

Feuille annexe, "Introduction pratique à l'analyse de Fourier", D. Oberson.