

## Outils Mathématiques pour le Traitement du signal

## TP1 – Analyse spectrale\*

CR à rendre dans une semaine, par binôme et dactylographié.

Étudier les questions 10 à 13 avant de commencer le TP.

L'analyse spectrale est l'étude du contenu fréquentiel d'un signal, fondée sur la transformation de Fourier. L'objectif est de mettre en œuvre, avec Matlab<sup>®</sup>, une partie de ces outils.

## 1 Simulation de signaux

On considère une sinusoïde à la fréquence  $f_0 = 440$  Hz

$$x(t) = A \sin(2\pi(f_0 t + \varphi)), \forall t \in \mathbb{R},$$

avec l'amplitude  $A \in \mathbb{R}$  et la phase réduite  $\varphi \in [0, 1]$ . C'est le « la » du diapason.

Ce signal est échantillonné aux instants  $t_n = nT_e$ , soit à la cadence  $F_e = 1/T_e = 10$  kHz, pour produire un signal numérique  $\{x_n\}_{n \in \mathbb{Z}}$  avec

$$x_n = A \sin(2\pi(\nu_0 n + \varphi)), \forall n \in \mathbb{Z},$$

où  $\nu_0$  est la fréquence réduite.

1. Que vaut  $\nu_0$  en fonction de  $f_0$  et  $F_e$  ?
2. Avec l'éditeur de Matlab (FILE → NEW), créer un fichier de commande (ou script) contenant la séquence de commande suivante

```
% efface les variables, ferme les figures
clear all, close all
```

```
f0 = 440;
Fe = 1e4;
nu0 = f0 / Fe;
phi = 0.27;
alpha = 1.2;
N = 1024;
```

**Remarque 1** Chaque lignes peut être indifféremment exécutée dans la console interactive ou écrite dans un script qui sera exécuté avec `TOOLS → RUN`. Pour cela il faut ajouter le dossier du script au `PATH` avec `FILE → SET PATH` pour que Matlab puisse le trouver.

3. Exécuter le programme et s'assurer que tout s'est passé comme attendu avec `whos`.

**Remarque 2** `whos` donne la liste des variables en mémoire. Pour obtenir la valeur d'une variable : taper son nom suivie de la touche `enter`.

4. Créer le signal avec

```
n = (0:N-1)';
sig = alpha * sin(2 * pi * (nu0 * n + phi));
```

**Remarque 3** La première ligne fabrique un vecteur contenant le temps discret. Le symbole `'` désigne l'opération transposée. La seconde est vectorielle : chaque élément du vecteur `n` est multiplié par la valeur `nu0`, la valeur de `phi` est ajoutée à chaque élément, etc. ; enfin, la fonction `sin` donne le sinus de chaque élément. Toutes les opérations sont vectorielles ou vectorisées.

5. Les vecteurs `n` et `sig` sont-ils des vecteurs lignes ou des vecteurs colonnes ?

6. Tracer le signal avec

```
figure(1), clf
plot(n, sig)
```

L'axe horizontal est gradué avec le numéro des échantillons.

**Remarque 4** Matlab dispose d'une aide en ligne en tapant `help` ou `doc`. Par exemple, `help plot` ou `doc plot` donne le manuel d'utilisation de la commande `plot`. La fin de chaque manuel renvoie à d'autres commandes sur le même sujet.

**Remarque 5** Sauvegarder les figures avec `FILE → EXPORT`.

7. Créer la variable `t`, à partir de `n`, contenant les instants temporels d'échantillonnage. Tracer, avec unités, `sig` en fonction du temps avec

```
figure(1), clf
subplot(2, 1, 1)
plot(t, sig)
subplot(2, 1, 2)
plot(t(10:100), sig(10:100))
```

**Remarque 6** Utiliser également les commandes `axis`, `grid`, `title`, `xlabel`. La syntaxe `t(10:100)` extrait les éléments du tableau `t` des indices 10 à 100.

**Remarque 7** Tous les graphiques doivent avoir les axes correctement labélisés.

8. Observer l'effet obtenu avec une fréquence d'échantillonnage  $F_e$  dix et cent fois plus élevée ; avec  $F_e$  dix et cent fois plus faible. Relier le résultat au phénomène de repliement spectral.

9. Ajouter à la variable `sig` une deuxième sinusoïde proche de la première, avec par exemple une fréquence  $f_1 = 460$  Hz et d'amplitude voisine. Ajouter enfin une troisième sinusoïde à 2,5 kHz et d'amplitude moitié. Tracer le résultat.

\*F. Orieux, M. Kowalski, T. Rodet, ...

## 2 Analyse spectrale

### 2.1 Rappels

10. Donner l'expression de la transformée de Fourier à temps continu du signal  $x(t)$  ?
11. Donner l'expression de la transformée de Fourier à temps discret du signal  $\{x_n\}_{n \in \mathbb{Z}}$  ?
12. Quel lien y-a-t-il entre elles ? Pourquoi est-on obligé de définir une transformée de Fourier discrète (TFD) à horizon fini ?
13. Sur quel interval de fréquence la TFD est calculée.

### 2.2 Analyse

14. En utilisant la fonction `fft`, calculer le vecteur `spectre` contenant la transformée de Fourier discrète (TFD) du vecteur `sig`.

**Remarque 8** Le résultat de `fft` peut être passé à `fftshift` pour faire comme si l'intervalle de fréquences (ou la période) était centré en 0.

15. Tracer la partie réelle, partie imaginaire, module au carré et phase du résultat de la TFD.

**Remarque 9** La fonction `real` renvoie la partie réelle. Le manuel de cette fonction renvoie aux fonctions permettant de déterminer la partie imaginaire, le module et la phase.

On met tous les éléments d'un vecteur  $\mathbf{v}$  au carré avec `v.^2`, ou avec la fonction `pow` comme `pow(v, 2)`.

16. Graduer correctement l'axe des fréquences en fréquence réelle.

**Remarque 10** Le vecteur de fréquence peut être fabriqué avec la fonction `linspace`.

17. Tracer le  $\log_{10}$  du module.
18. Retrouver les « pics » aux fréquences 440 Hz, 460 Hz et 2,5 kHz. Sont-ils positionnés précisément ? Évaluer cette précision en fonction de  $N$ . Expliquer la présence des trois autres pics.
19. Constater des « rebonds » au pieds des pics. On parle de *ringing* dans la littérature anglosaxonne. Expliquer leur origine.
20. Observer les composantes à 440 et 460 Hz lorsque  $N = 1024$ ,  $N = 512$  et  $N = 256$ . Commenter.
21. Commenter les résultats obtenus avec  $F_e = 6$  kHz,  $F_e = 4$  kHz et  $F_e = 0,5$  kHz ?

### 2.3 Analyse d'un signal audio

Vous avez également la possibilité d'enregistrer un son avec le microphone.

```
% Pour enregistrer le micro en 8 bits à 8 kHz
recObj = audiorecorder;
% Enregistre le son pendant 5 secondes
recordblocking(recObj, 5);
% Récupère le son enregistré
snd_data = getaudiodata(recObj);
```

La variable `snd_data` est un vecteur contenant le signal enregistré du micro.

**Remarque 11** Si `audiorecorder` ne fonctionne pas, ou si il n'y a pas de micro, vous pouvez vous enregistrer avec votre téléphone et lire le fichier avec `audioread`. En dernier recourt, utilisez le fichier « `sifflement.wav` ».

La command `audiowrite` permet d'écrire un fichier et on peut l'écouter avec `sound`.

Utilisez par exemple un « sifflement » pour avoir un signal avec une fréquence précise. Quelle est sa fréquence ? En binôme ou en trinôme vous pouvez aussi enregistrer simultanément plusieurs sifflements et identifier la fréquence de chacun d'entre eux.