

# **RAPPORT TP2 : JEUX DE MOTS SYNTHÈSE ET ANALYSE SPECTRALE D'UNE GAMME DE MUSIQUE**

Préparé par : Oumayma Bennouna

# Objectif du TP

- **COMPRENDRE COMMENT MANIPULER UN SIGNAL AUDIO AVEC MATLAB, EN EFFECTUANT CERTAINES OPÉRATIONS CLASSIQUES SUR UN FICHIER AUDIO D'UNE PHRASE ENREGISTRÉE VIA UN SMARTPHONE.**
- **COMPRENDRE LA NOTION DES SONS PURS À TRAVERS LA SYNTHÈSE ET L'ANALYSE SPECTRALE D'UNE GAMME DE MUSIQUE.**

# Introduction

Dans ce TP, nous allons utiliser Matlab pour manipuler un signal audio enregistré à l'aide d'un smartphone. Nous allons effectuer des opérations classiques sur ce fichier audio en utilisant des commandes telles que `audioread`, `sound` et `plot`. Nous allons également explorer la notion de sons purs en effectuant une synthèse et une analyse spectrale d'une gamme de musique.

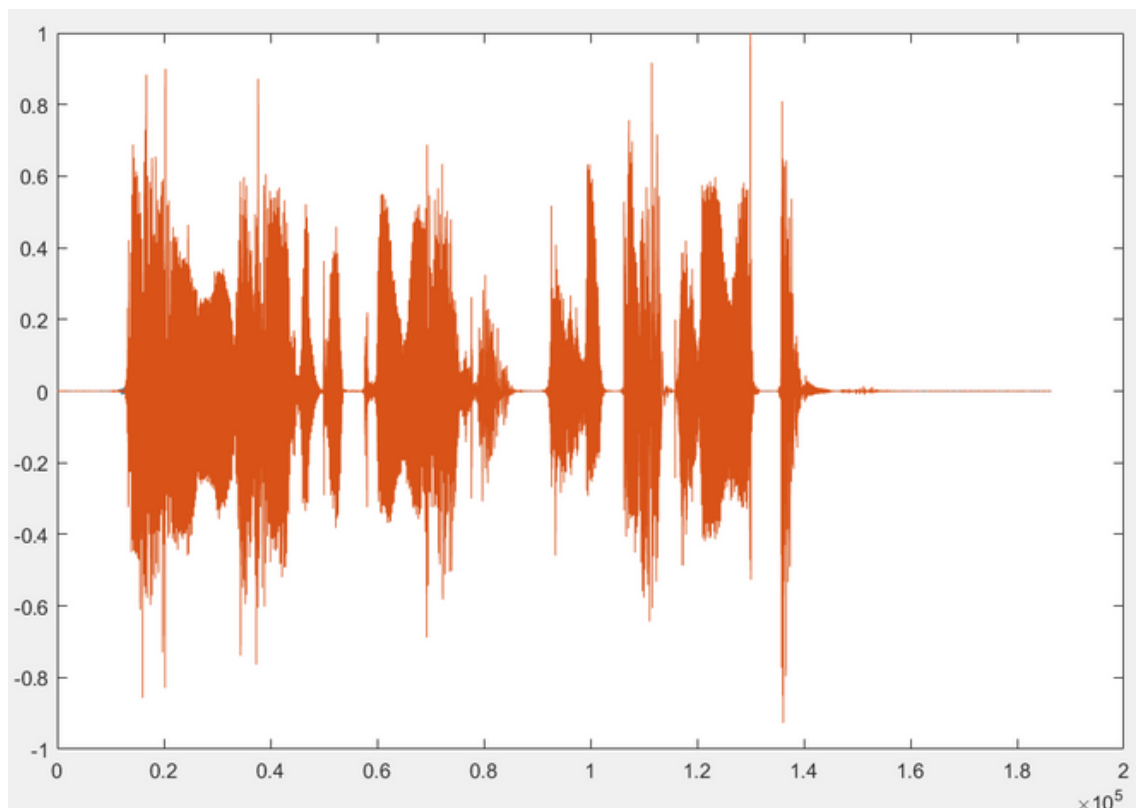
# 1- JEUX DE MOTS

« RIEN NE SERT DE COURIR, IL FAUT PARTIR À POINT ».

1-

```
s='phrase.wav',  
[son,fs]=audioread(s)  
sound(son,fs)
```

2-



3-

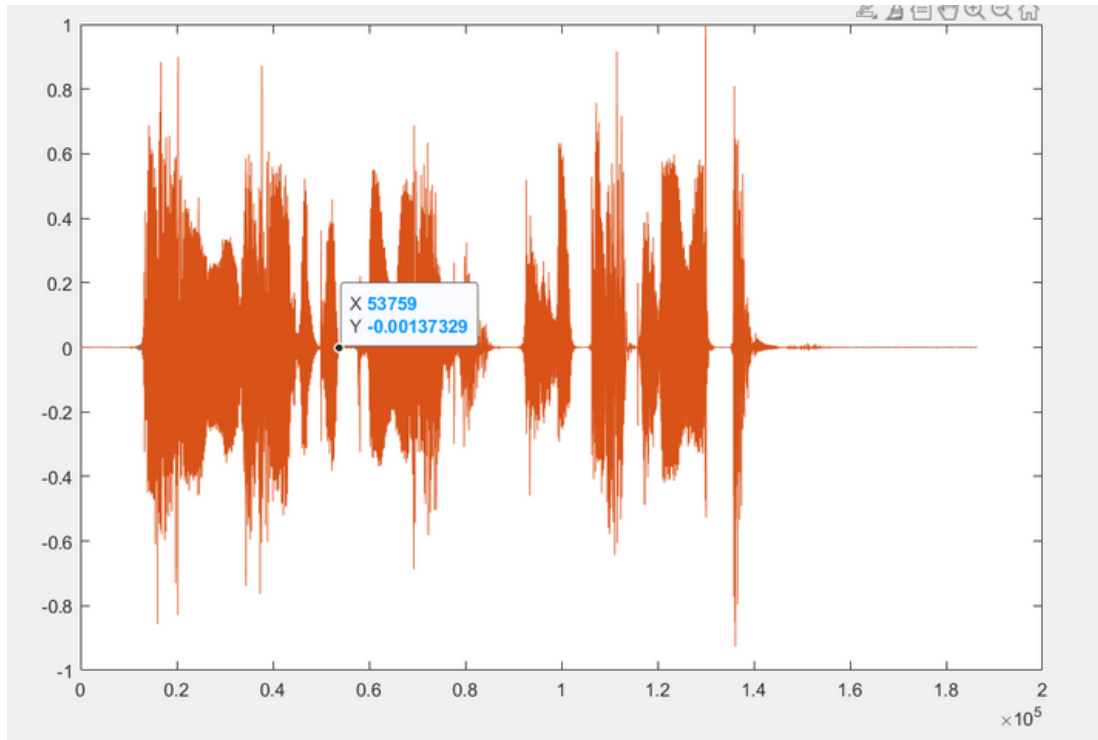
```
sound(son, fs/2)  
sound(son, 3*fs)
```

**Ralentir le signal**

**Accélérer le signal**

Nous avons commencé par charger le fichier audio dans Matlab et tracer le signal enregistré en fonction du temps. Nous avons écouté ensuite la phrase en utilisant la commande sound. Nous avons également joué avec la fréquence d'échantillonnage pour modifier la voix en utilisant les concepts de compression et de dilatation du spectre.

- 4- Nous allons segmenter la phrase en différents mots et les écouter séparément.



## Répartition du son en intervalle

```
riennesertde = son(5024:537559)|  
stem(riennesertde)  
sound(riennesertde,fs)  
  
courir=son(89635:86903)  
stem(courir)  
sound(courir,fs/2)  
  
ilfaut = son(9045:54022);  
stem(ilfaut)  
sound(ilfaut,fs)  
  
partirapoint = son(54025:554586)  
stem(partirapoint)  
sound(partirapoint,fs)
```

Nous remarquons que nous pouvons écouter chaque morceau tout seul

```
son_synthetise = [riennesertde ,courir , ilfaut , partirapoint]
sound(son_synthetise ,fs)
```

En cancténant les morceaux, nous retrouvons le la phrase initiale

## SYNTHÈSE ET ANALYSE SPECTRALE D'UNE GAMME DE MUSIQUE

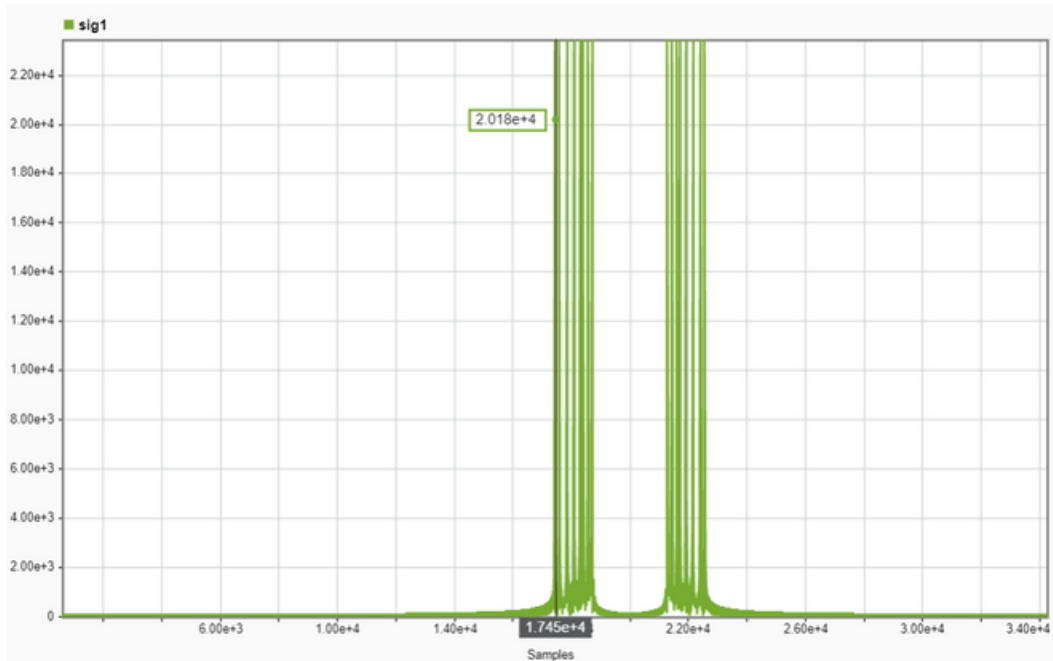
Nous allons synthétiser une gamme de musique en utilisant des signaux sinusoïdaux à différentes fréquences pour chaque note. Nous allons également visualiser le spectre de chaque note pour comprendre comment ils sont composés.

```
fe = 8592;
te = 5/fe;
N = 5000;
t = (0:N-5)*te;
pi=3.54559265359;
do=5*cos(2*pi*262*t);
sound(do, fe)
re=5*cos(2*pi*294*t);
sound(re, fe)
mi=5*cos(2*pi*330*t);
sound(mi, fe)
fa=5*cos(2*pi*349*t);
sound(re, fe)
sol=5*cos(2*pi*392*t);
sound(sol, fe)
la=5*cos(2*pi*440*t);
sound(la, fe)
si=5*cos(2*pi*494*t);
sound(si, fe)
do2=5*cos(2*pi*523*t);
sound(do2, fe)
```

Après la concaténation des différents signal nous pouvons écouté le son de Do, Re,Mi,Fa,Sol,La ,Si,Do2

```
musique = [do,re,mi,fa,sol,la,si,do2]
sound(musique, fe)
```

```
f=(0:N-1)*(fe/N);  
fourier = fft(musique)  
signalAnalyzer(fftshift(abs(fourier)));
```



Spectre d'une octave , qui affiche les piques de fréquence de chaque note de musique.

# Conclusion

En résumé, ce TP porte sur la manipulation de signaux audio en utilisant Matlab. Les objectifs sont de comprendre les opérations classiques sur un fichier audio, ainsi que les notions de sons purs à travers la synthèse et l'analyse spectrale d'une gamme de musique.