

## TP

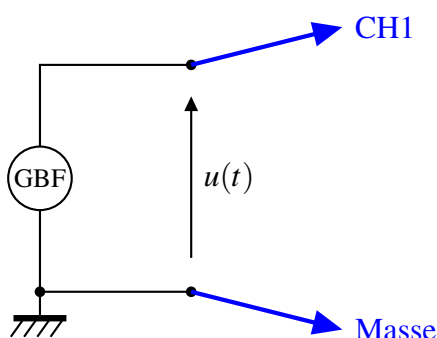
# Décomposition spectrale d'un signal

## Introduction

Ce TP a pour objectif d'introduire l'analyse fréquentielle d'un signal périodique. Il propose d'étudier la décomposition spectrale d'un signal carré.

## Montage

La tension à étudier est fournie par un générateur basse fréquence (GBF).



$u(t)$  est une tension carrée évoluant strictement entre  $+6\text{ V}$  et  $-4\text{ V}$  avec une fréquence de  $1,0\text{ kHz}$ .

## Évolution temporelle

**Q1.** Relever l'évolution temporelle de la tension  $u(t)$ .

Y faire apparaître les mesures de sa période  $T$ , de sa fréquence  $f$ , de son amplitude crête à crête  $A_{pp}$  et de sa valeur moyenne  $\langle u \rangle$ .

Mettre la copie d'écran en **annexe 1**.

## Spectre d'amplitude

A partir du programme `prog_1.py` à compléter (mettre en **annexe 2**) :

- Q2.** Tracer l'évolution temporelle du signal  $u(t)$  (courbe en **annexe 2**).
- Q3.** En déduire le tracé de son spectre d'amplitude (courbe en **annexe 2**).
- Q4.** Donner les valeurs des amplitudes pour les 10 premières fréquences du spectre d'amplitude.
- Q5.** Qu'est-il possible de conclure pour les harmoniques de rang pair d'un signal carré ?
- Q6.** En déduire, l'expression de la décomposition harmonique de la tension  $u(t)$  sous forme numérique pour ces 10 premières fréquences.

Pour rappel, la décomposition harmonique d'un signal périodique est toujours de la forme :

$$u(t) = A_0 + A_1 \cdot \sin(\omega t + \varphi_1) + A_2 \cdot \sin(2\omega t + \varphi_2) + A_3 \cdot \sin(3\omega t + \varphi_3) + \dots$$

De plus, pour un signal carré, toutes les phases sont nulles !

**Q7.** À quelle valeur particulière correspond le paramètre  $A_0$  ?

**Q8.** Vérifier maintenant que :

$$A_1 = \frac{4 \cdot A}{\pi} \quad \text{avec} \quad A = \frac{A_{pp}}{2}$$

**Q9.** Identifier la relation générale entre l'amplitude  $A_n$  de l'harmonique de rang  $n$  et l'amplitude  $A_1$  du fondamental ?

## Conclusion

**Q10.** Que retenir de la décomposition spectrale d'un signal carré ?

## Synthèse harmonique de la tension

Il s'agit ici de reconstituer la tension  $u(t)$  à partir de sa décomposition harmonique trouvée précédemment.

**Q11.** À partir du programme `prog_2.py` fourni, tracer l'allure de la tension  $u(t)$  jusqu'au rang 5.

**Q12.** Compléter ce programme pour obtenir les allures de la tension  $u(t)$  jusqu'au rang 7, 9, ...

**Q13.** Donner une formule générale de la décomposition spectrale d'un signal carré en utilisant une somme infinie sous la forme :

$$\sum_{n=0}^{\infty} (\dots)$$

Mettre le programme complété et les courbes obtenues en **annexe 3**.

# Programme 1

```
# prog_1.py
import numpy as np
import matplotlib.pyplot as plt
from physapp import spectre_amplitude

#### IMPORTATION DES DONNEES CSV ####
t, u = np.loadtxt('scope.csv', delimiter=',', skiprows=2, unpack=True)

#### CALCUL DU SPECTRE D'AMPLITUDE ####
f, Amp = spectre_amplitude(???)      # spectre_amplitude(y, t, period=None, tmin=None, plot_period_ax=None)

#### COURBE DE u(t) ####
plt.figure()
plt.plot(???)                        # Trace la courbe u(t)
plt.title("???")                     # Titre
plt.xlabel('???')                    # Légende en abscisse
plt.ylabel('???')                    # Légende en ordonnée
plt.grid()                           # Dessine une quadrillage
plt.savefig("courbe_1.png")           # Sauvegarde en PNG

#### COURBES ####
plt.figure()
plt.stem(???)                        # Trace le spectre sous forme de raies (basefmt="gray")
plt.title("???")                     # Titre
plt.xlabel('???')                    # Légende en abscisse
#plt.xlim(min, max)                  # Echelle sur l'axe des x
plt.ylabel('???')                    # Légende en ordonnée
plt.grid()                           # Dessine une quadrillage
plt.savefig("courbe_2.png")           # Sauvegarde en PNG

#### AFFICHAGE DES COURBES ####
plt.show()                           # Affichage
```

Aide de la fonction `spectre_amplitude` de la librairie `physapp` :

```
>>> help(spectre_amplitude)
Help on function spectre_amplitude in module physapp.base:

spectre_amplitude(y, t, period=None, tmin=None, plot_period_ax=None)
    Retourne le spectre d'amplitude d'un signal y(t).

    Parameters
    -----
    y : numpy.ndarray
        Tableau des valeurs du signal.

    t : numpy.ndarray
        Tableau des temps.

    T : float
        Période du signal.

    tmin : float, optionnel (0 par défaut)
        Borne inférieure le calcul du spectre.

    plot_period_ax : matplotlib.axes, optionnel (None par défaut)
```

Repère (axes) sur lequel tracer la sélection de la période.

Return

-----

(f, A) : (numpy.ndarray, numpy.ndarray)

Tableaux des fréquences et des amplitudes.

## Programme 2

```
# prog_2.py
import numpy as np
import matplotlib.pyplot as plt
from math import pi

#### CALCULS ####
T = ???                # Période en s
w = ???                # Pulsation
t = np.linspace(0, T, 1000) # Tableau des temps
u = ???                # Tension à calculée

#### COURBES ####
plt.plot(t, u, label="Synthèse") # Trace la tension
plt.legend()
plt.title("???")                # Titre
plt.xlabel('???')                # Légende en abscisse
plt.ylabel('???')                # Légende en ordonnée
plt.grid()                      # Grille

#### AFFICHAGE DES COURBES ####
plt.savefig("courbe_3.png")      # Sauvegarde en PNG
plt.show()                      # Affichage de la courbe
```