1 Rappels

Toutes les questions de ces TDs ne seront pas résolues en séance. Vous êtes invités à essayer de les résoudre (tant que faire se peut ...).

Certains exercices sont issus des cours “discrete-time signal processing” de Oppenheim, Schafer et Buck.

Table de transformées en z

$x[n]$	$X(z)$	Région de convergence (ROC)
$\delta[n]$	1	$z \in \mathbb{C}$
$u[n]$	$\frac{z}{z-1}$	$\lvert z\rvert > 1$
$(-a)^n u(n)$	$\frac{z}{z+a}$	$\lvert z\rvert > a$
$nu[n]$	$\frac{z}{(z-1)^2}$	$\lvert z\rvert > 1$
$n^2 u[n]$	$\frac{z(z+1)}{(z-1)^3}$	$\lvert z\rvert > 1$
$e^{an}u[n]$	$\frac{z}{z-e^a}$	$\lvert z\rvert > \lvert e^a\rvert$
$C_{k-1}^{n-1} e^{a(n-k)}u[n-k]$	$\frac{z}{(z-e^a)^k}$	$\lvert z\rvert > \lvert e^a\rvert$
$\cos(\omega n)u[n]$	$\frac{z(z-\cos(\omega))}{z^2-2z\cos(\omega)+1}$	$\lvert z\rvert > 1$
$\sin(\omega n)u[n]$	$\frac{z\sin(\omega)}{z^2-2z\cos(\omega)+1}$	$\lvert z\rvert > 1$
$\frac{1}{n}u[n-1]$	$\ln\left(\frac{z}{z-1}\right)$	$\lvert z\rvert > 1$
$\sin(\omega n+\theta)u[n]$	$\frac{z^2\sin(\theta)+z\sin(\omega-\theta)}{z^2-2z\cos(\omega)+1}$	$\lvert z\rvert > 1$
$e^{an}\cos(\omega n)u[n]$	$\frac{z(z-e^a\cos(\omega))}{z^2-2ze^a\cos(\omega)+e^{2a}}$	$\lvert z\rvert > \lvert e^a\rvert$ height

1.1 Réponses fréquentielles (en amplitude et en phase)

Calculez et tracez les réponses fréquentielles des systèmes décrits par les équations aux différences suivantes :

- $y[n] = x[n] + 2x[n-1] + 3x[n-2] + 2x[n-3] + x[n-4]$
- $y[n] = y[n-1] + x[n]$
- $y[n] = x[n] + 3x[n-1] + 2x[n-2]$

1.2 Système LTI

Soit le filtre numérique exprimé par

$$H(z) = \frac{1 + 2.05.z^{-1} - 2.85.z^{-2}}{(1 + 0.9z^{-2})(1 + 0.95z^{-1})}$$

Quel est le type de ce filtre (IIR/FIR). Donnez son équation de différences.

Trouvez les pôles et les zéros et représentez-les sur le plan complexe.

Donnez l'expression de $H(f)$.

Tracez le gain du filtre $|H(f)|$ entre $f = -/2$ et $f = 1/2$, pour 5 valeurs positives de f.

Justifiez ce tracé en utilisant le diagramme des pôles et zéros.

1.3 Conception d'un filtre simple

Concevez un système LTI causal (à temps discret) avec les propriétés suivantes :

- Le système doit préserver exactement le signal $\cos(0.5\pi n)$.
- Le système doit annihiler les signaux constants (réponse fréquentielle nulle à la fréquence 0).

Pour ce système, donnez l'équation aux différences (il est possible d'avoir une réponse impulsionnelle de longueur 3), tracez la réponse fréquentielle, le diagramme des pôles et zéros.

1.4 Paramètres d'un filtre

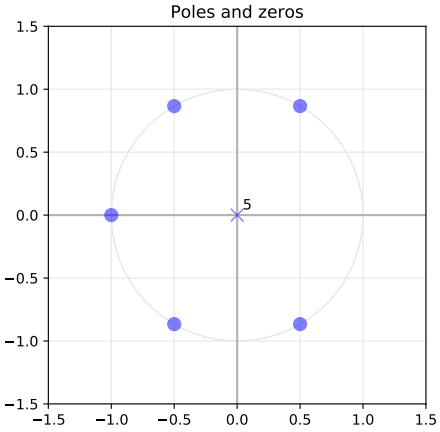
Soit un filtre donné par la fonction de transfert suivante :

$$H(z) = \frac{\delta_o + \delta_1 z^{-1} - \delta_2 z^{-1} f(z)}{1 - (1 + m_1)z^{-1} - m_2 z^{-1} f(z)}$$

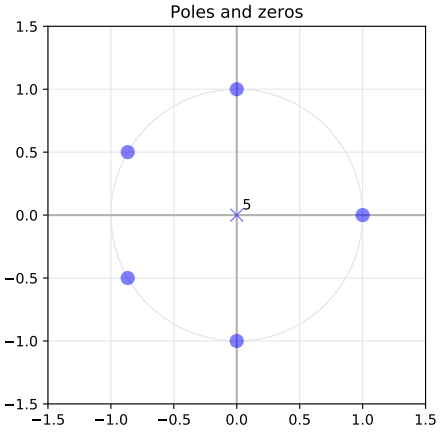
où $f(z) = \frac{1}{1-z^{-1}}$. Faites la conception du filtre de telle sorte qu'il ait un gain unitaire en courant continu et un zéro double (deux zéros) à $\omega = \pi$.

1.5

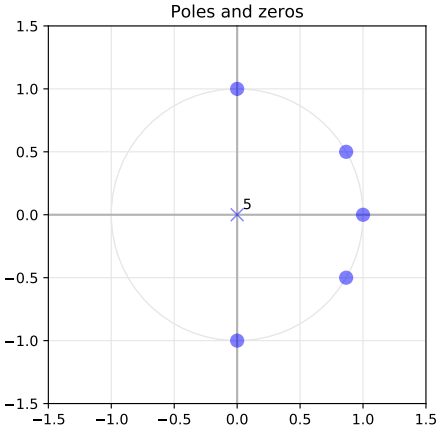
Soient les 6 filtres (FIR) suivants, donnez les correspondances entre les figures (pole/zéro - réponse impulsionnelle - réponse fréquentielle) ,


Poles and zeros
5

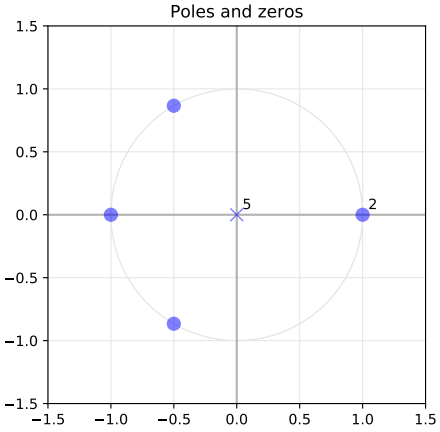
(1)


Poles and zeros
5

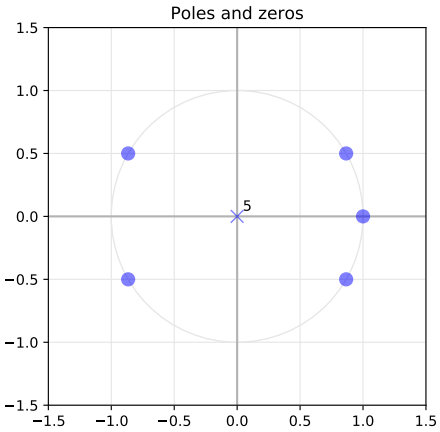
(2)


Poles and zeros
5

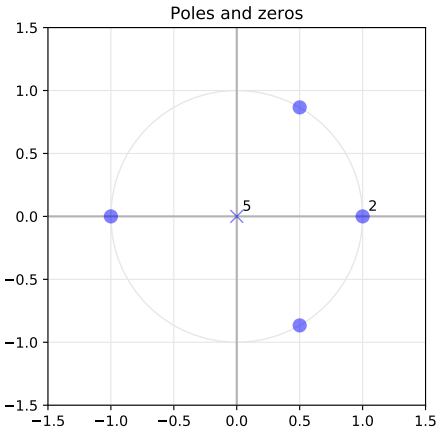
(3)


Poles and zeros
5
2

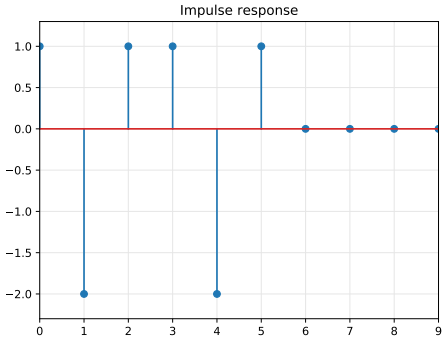
(4)


Poles and zeros
5

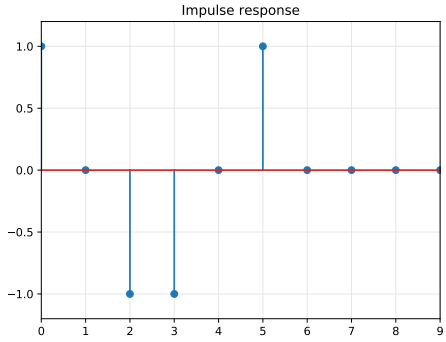
(5)


Poles and zeros
5
2

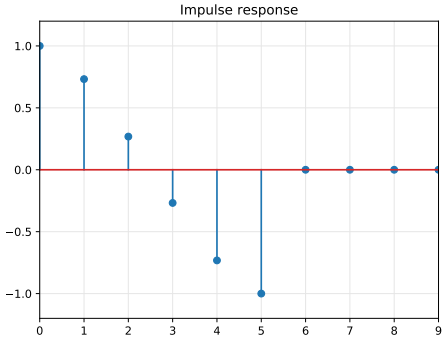
(6)


Impulse response

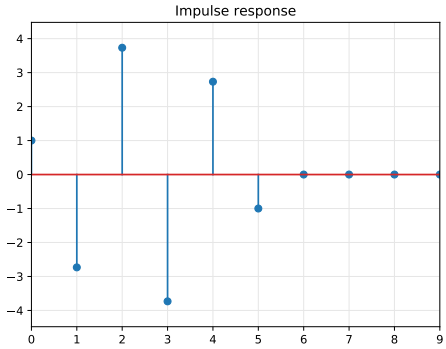
(1)


Impulse response

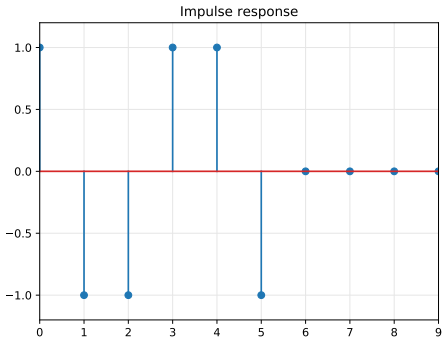
(2)


Impulse response

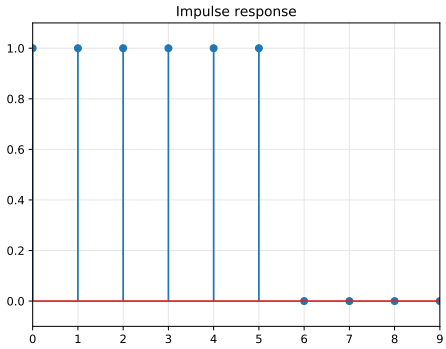
(3)


Impulse response

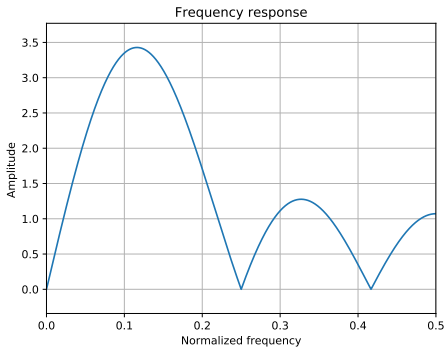
(4)


Impulse response

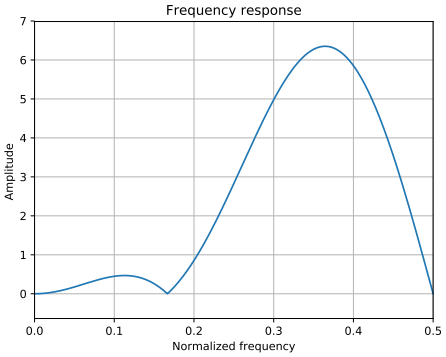
(5)


Impulse response

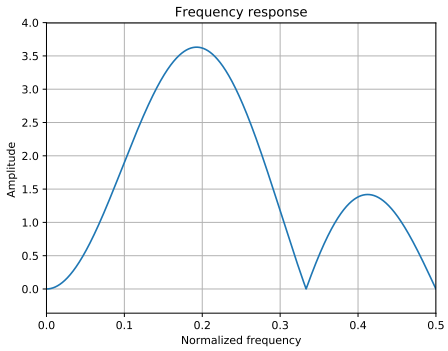
(6)


Frequency response
Amplitude
Normalized frequency

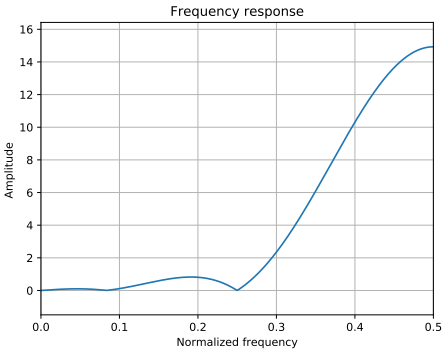
(1)


Frequency response
Amplitude
Normalized frequency

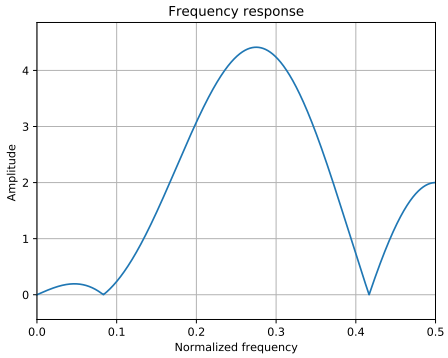
(2)


Frequency response
Amplitude
Normalized frequency

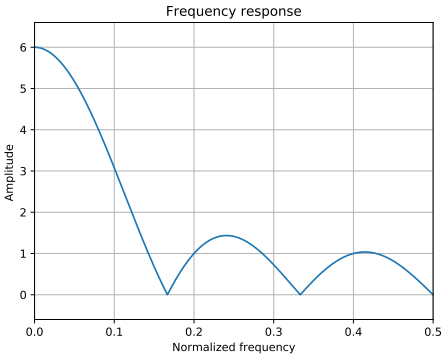
(3)


Frequency response
Amplitude
Normalized frequency

(4)


Frequency response
Amplitude
Normalized frequency

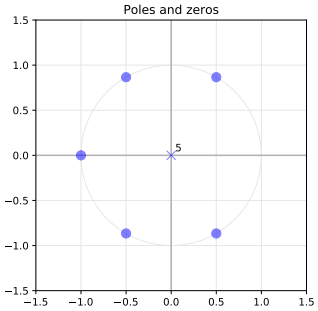
(5)


Frequency response
Amplitude
Normalized frequency

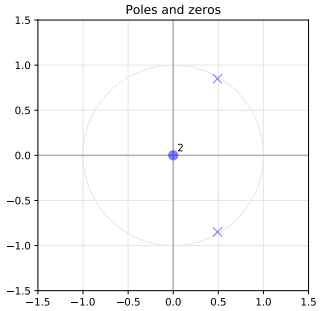
(6)

1.6

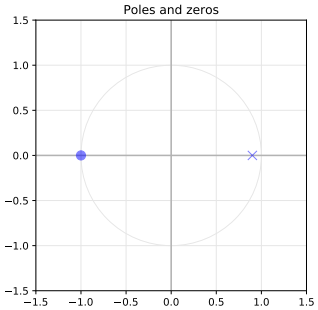
Soient les 6 filtres suivants, donnez les correspondances entre les figures (pole/zéro - réponse impulsionnelle - réponse fréquentielle) ,



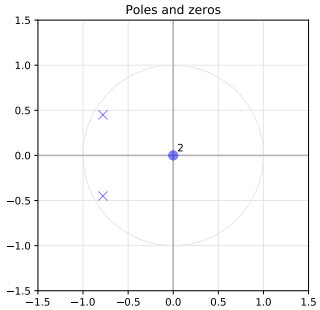
(1)



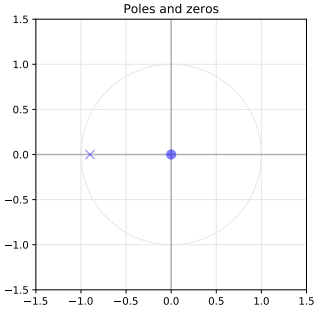
(2)



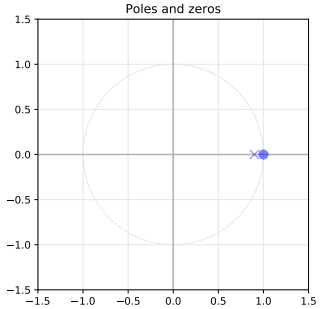
(3)



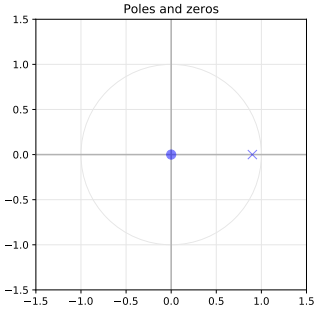
(4)



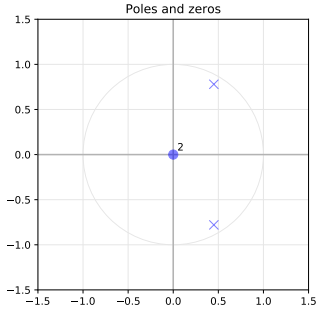
(5)



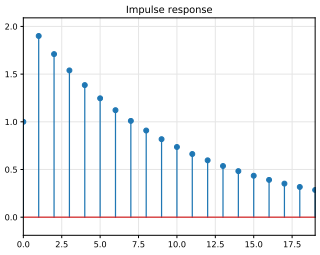
(6)



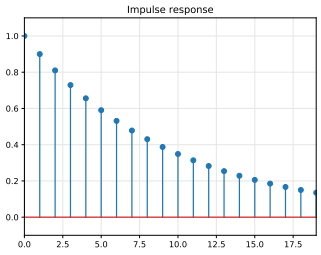
(7)



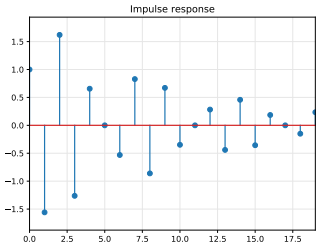
(8)


Impulse response

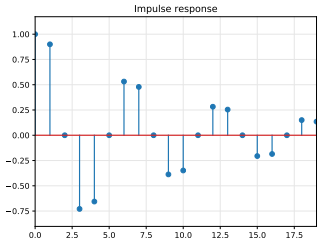
(1)


Impulse response

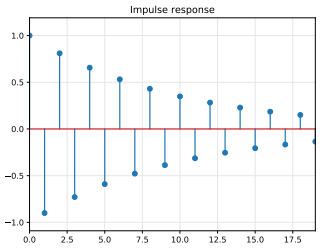
(2)


Impulse response

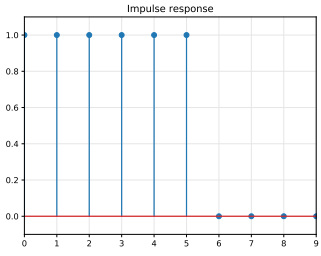
(3)


Impulse response

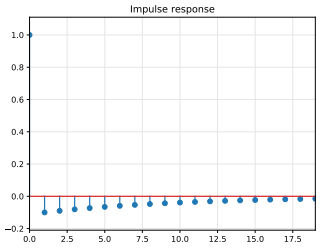
(4)


Impulse response

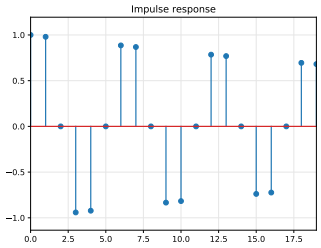
(5)


Impulse response

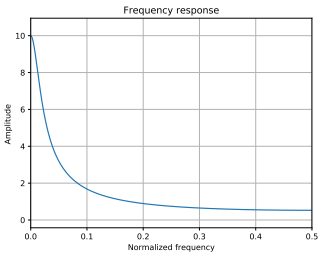
(6)


Impulse response

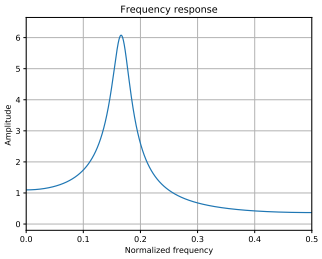
(7)


Impulse response

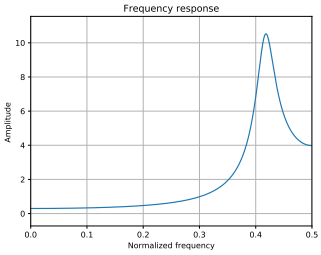
(8)


Frequency response
Amplitude
Normalized frequency

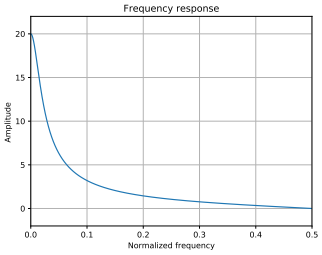
(1)


Frequency response
Amplitude
Normalized frequency

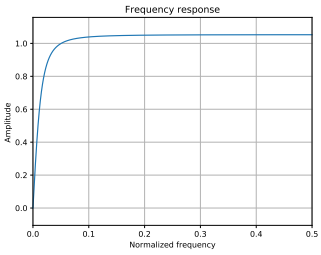
(2)


Frequency response
Amplitude
Normalized frequency

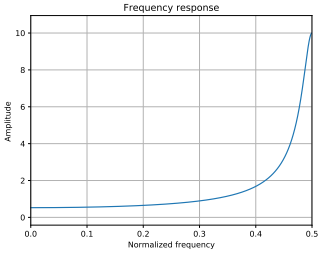
(3)


Frequency response
Amplitude
Normalized frequency

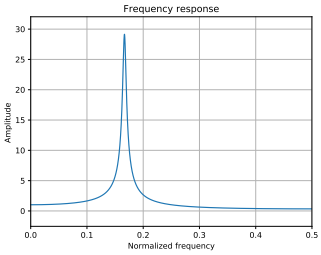
(4)


Frequency response
Amplitude
Normalized frequency

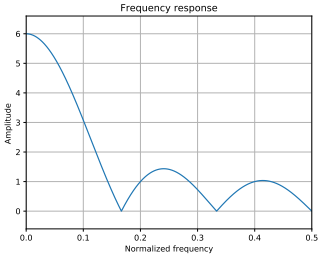
(5)


Frequency response
Amplitude
Normalized frequency

(6)


Frequency response
Amplitude
Normalized frequency

(7)


Frequency response
Amplitude
Normalized frequency

(8)

2 Conception de filtres FIR

Les exercices suivant sont à faire avec Python (sur Spyder ou Jupyter).

2.1 Comparaison des fenêtres

Concevez un ensemble de filtres passe-bas de fréquence de coupure $\omega_c = \pi/4$, d'ordre 11. Vous utiliserez les fenêtres rectangulaire, de Hamming, de Hanning, de Blackmann et de Blackmann Harris. Comparez les différentes caractéristiques de ces filtres.

Modifiez l'ordre du filtre, quel en est l'effet ?

Exprimez les pentes des zones de transition en dB/octave ou dB/décade.

2.2 Elimination du bourdonnement à 100 Hz

Un problème classique dans les enregistrements audio de mauvaise qualité, où dans les transmissions sur supports analogiques, est la présence d'un bourdonnement à 50 Hz. Pour l'exercice nous utiliserons un bourdonnement à 100 Hz.

Dans un premier temps, vous allez charger un fichier musical de votre choix, le décimer (passer de 44 kéch/sec à 5.5 kéchs/sec), additionner un signal sinusoïdal à 100 Hz et ... écouter le résultat.

Dans un deuxième temps, vous allez concevoir un filtre permettant d'éliminer ce bruit, en essayant de détériorer le signal original le moins possible. Comparez les deux résultats.

2.3 Elimination du battement

Quand on additionne deux signaux sinusoïdaux de fréquence proche, apparaît un battement. En effet :

$s(t) = \sin(\omega_1 t) + \sin(\omega_2 t) = 2\sin(\frac{\omega_1 + \omega 2}{2}t)\cos(\frac{\omega_1 - \omega 2}{2}t)$

et donc $s(t)$ est un signal à la moyenne des fréquences, modulé par un signal à la (moitié de la) différence des fréquences, appelé battement.

Dans cet exercice, on travaillera dans un premier temps avec une $f_1 = 880Hz, f_2 = 884Hz$. On déterminera un ordre de filtre FIR par "kaiserord", et on fera le design du filtre avec l'algorithme de Remez.

Commentez la longueur du filtre, et donnez sa complexité (en nombre de multiplications par seconde nécessaires).

Exemple de code pour générer un battement :

```
%matplotlib inline
import numpy as np
import scipy.signal as signal
import matplotlib.pyplot as plt
import numpy as np
import IPython



Fs=8000

f1=880
```

```
f2=884
om1=2*np.pi*f1
om2=2*np.pi*f2
t=np.arange(0,5,1/Fs)
y=np.sin(om1*t)+np.sin(om2*t)


IPython.display.Audio(y, rate=Fs)
```