

Etude des modes quasi-normaux d'une corde vibrante

Comment le transfert d'énergie aboutissant à la production d'un son par un instrument à cordes se fait-il ? Comment caractériser et formaliser les fuites d'énergie ?

MOULIN Hugues - 25398

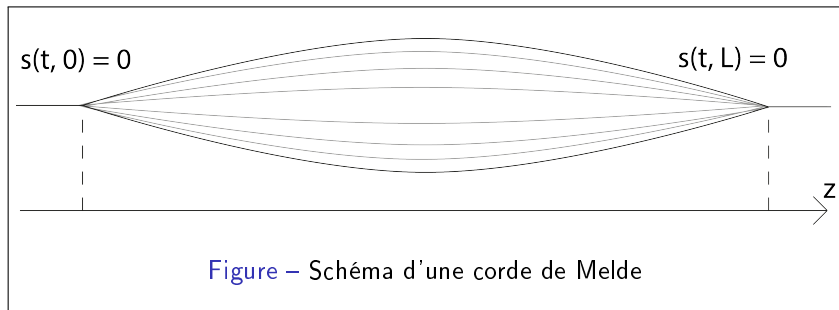
SOMMAIRE

- 1 Introduction : Corde de Melde
- 2 Cavité semi-ouverte
 - Système
 - Nécessité d'une pulsation complexe
 - Modes quasi-normaux et projection de l'énergie
- 3 Corde Quasi-normale
 - Expression des ondes
 - Expérience
 - Effet supplémentaire
- 4 Conclusion

Introduction : Corde de Melde

Dans une cavité fermée : $s(t, z) = A \cos(\omega_n t) \cos(k_n z)$

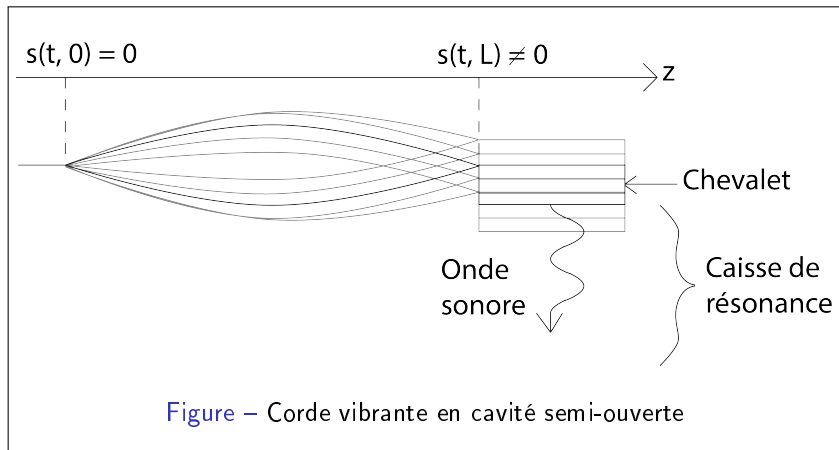
avec $\begin{cases} \omega_n = n\omega_0 \\ k_n = nk_0 \end{cases} \quad n \in \mathbb{N}, (\omega_0, k_0) \in \mathbb{R}^2$



- 1 Introduction : Corde de Melde
- 2 Cavité semi-ouverte
 - Système
 - Nécessité d'une pulsation complexe
 - Modes quasi-normaux et projection de l'énergie
- 3 Corde Quasi-normale
 - Expression des ondes
 - Expérience
 - Effet supplémentaire
- 4 Conclusion

Système

Cavité ouverte en L



N cessit  d'une pulsation complexe

r coefficient de reflexion tel que $r \in]0, 1[$

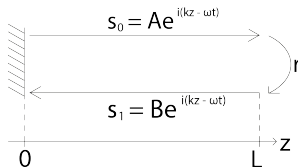


Figure – Onde dans une cavit  semi-ouverte

Continuit  en $t = \frac{L}{c}$:
 $rAe^{i(kz - \omega \frac{L}{c})} = Be^{i(kz + \omega \frac{L}{c})}$
 D'o  $B = rAe^{-2i(\omega \frac{L}{c})}$

De m me en $t = 0$: $A = B$
 Donc au final $1 = re^{-2i(\omega \frac{L}{c})}$ (★)
 Donc $|e^{-2i(\omega \frac{L}{c})}| > 1$

Condition sur ω

$$\omega \in \mathbb{C}$$

Nécessité d'une pulsation complexe

Expression de la pulsation

On pose $\omega = \omega^R + \omega^I$

D'après (★) on obtient

$$\begin{cases} e^{-2i\omega^R \frac{L}{c}} = 1 \\ e^{2\omega^I \frac{L}{c}} = \frac{1}{r} \end{cases}$$

Expression des pulsations

$$\begin{cases} \omega^R = \frac{k\pi c}{L} & k \in \mathbb{Z} \\ \omega^I = \frac{c}{2L} \ln\left(\frac{1}{r}\right) \end{cases}$$

Modes quasi-normaux et projection de l'énergie

Définition

Définition

On note E l'espace vectoriel des fonctions 2π -périodiques

On appelle modes normaux associés à ω_0 l'ensemble

$$\mathcal{S} = \{t \mapsto e^{ik\omega_0 t} | k \in \mathbb{N}\} \text{ avec } \omega_0 \in \mathbb{R}$$

Alors $\mathcal{S} \subset E$

On appelle modes quasi-normaux associés à ω l'ensemble

$$\mathcal{Q} = \{t \mapsto e^{ik\omega t} | k \in \mathbb{N}\} \text{ avec } \omega = \omega^R + i\omega^I \in \mathbb{C}$$

Modes quasi-normaux et projection de l'énergie

Cas des modes normaux

Dans une cavité fermée : on a $\omega_0 \in \mathbb{R}$

Signaux : $s_k : t \mapsto e^{ik\omega_0 t}$ avec $k \in \mathbb{N}$ donc $s_k \in \mathcal{S}$

Projection $\rightarrow$ produit scalaire

Produit scalaire utilisé ici sur E

$$\forall (f, g) \in E, \langle f | g \rangle = \frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{g(t)} dt$$

$$\text{Alors } \forall (s_n, s_p) \in \mathcal{S}, \langle s_n | s_p \rangle = \begin{cases} 0 & \text{si } n \neq p \\ 1 & \text{si } n = p \end{cases}$$

Conséquence sur les signaux

$(s_k)_{k \in \mathbb{N}} \in \mathcal{S}^{\mathbb{N}}$ famille orthonormale

Modes quasi-normaux et projection de l'énergie

Cas des modes quasi-normaux

Dans une cavité semi-ouverte : on a $\omega \in \mathbb{C} = \omega^I + i\omega^R$

Les signaux sont $s_k : t \mapsto e^{ik\omega t} \in \mathcal{Q}$

Dans ce cas

$$\forall (s_n, s_p) \in \mathcal{Q}^2, \langle s_n | s_p \rangle = \frac{1}{2\pi} \frac{e^{-2\pi(n+p)\omega^I} - 1}{-(n+p)\omega^I + i(n-p)\omega^R}$$

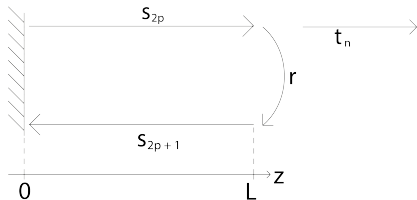
Conséquence sur les signaux

$(s_k)_{k \in \mathbb{N}} \in \mathcal{Q}^{\mathbb{N}}$ ni orthonormale ni orthogonale

- 1 Introduction : Corde de Melde
- 2 Cavité semi-ouverte
 - Système
 - Nécessité d'une pulsation complexe
 - Modes quasi-normaux et projection de l'énergie
- 3 Corde Quasi-normale
 - Expression des ondes
 - Expérience
 - Effet supplémentaire
- 4 Conclusion

Expression des ondes

Expression mathématique



On a $s_0(t) = Ae^{i(\omega t - kz)}$

Alors
$$\begin{cases} s_1 = rAe^{i(\omega t + kz)} \\ t_0 = 0 \\ t_1 = (1 - r)Ae^{i(\omega t - kz)} \end{cases}$$

Ainsi on obtient

$$\begin{cases} s_{2p}(t) = r^p Ae^{i(\omega t - kz)} \\ s_{2p+1}(t) = r^{p+1} Ae^{i(\omega t + kz)} \\ t_{2p}(t) = 0 \\ t_{2p+1}(t) = (1 - r)^{p+1} Ae^{i(\omega t - kz)} \end{cases}$$

Donc on pose

$$\vec{s}(t) = \sum_{p=0}^{+\infty} s_{2p}(t) = \frac{A}{1 - r} e^{i(\omega t - kz)}$$

$$\overleftarrow{s}(t) = \sum_{p=0}^{+\infty} s_{2p+1}(t) = r \frac{A}{1 - r} e^{i(\omega t + kz)}$$

$$t(t) = \sum_{p=0}^{+\infty} t_{2p+1}$$

$$s(t) = \vec{s}(t) + \overleftarrow{s}(t)$$

Signaux totaux

$$\begin{cases} s(t) = \frac{Ae^{i\omega t}}{1 - r} (e^{-ikz} + re^{ikz}) \\ t(t) = \frac{(1 - r)A}{r} e^{i(\omega t - kz)} \end{cases}$$

Vrai pour $\omega \in \mathbb{R}$ ou $\mathbb{C}$

Expression des ondes

Représentation pour une corde classique

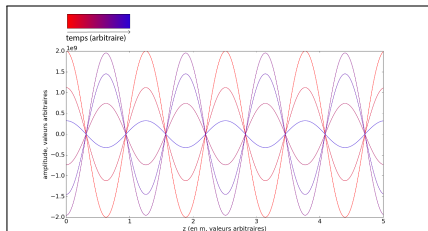


Figure – Onde stationnaire non amortie

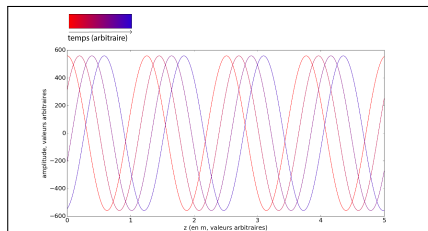


Figure – Onde progressive transmise non amortie

Expression des ondes

Représentation pour une corde amortie

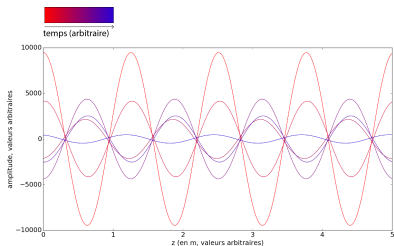


Figure – Onde stationnaire amortie

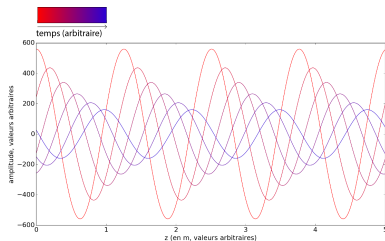
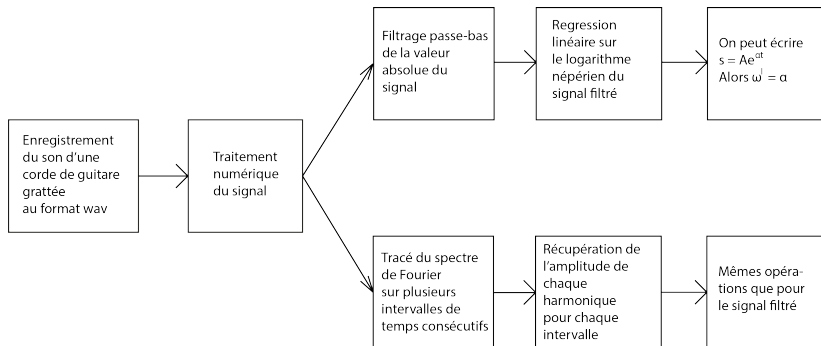


Figure – Onde progressive transmise amortie

Expérience

Protocole



Expérience

Résultats pour toutes les cordes

Pour le fondamental des différentes cordes de la guitare

Corde	Mi grave	La	Ré
ω^R (en Hz)	508 ± 6	680 ± 40	900 ± 20
ω^I (en s^{-1})	-0.3 ± 0.1	-0.55 ± 0.07	-0.6 ± 0.1

Corde	Sol	Si	Mi aigu
ω^R (en Hz)	1200 ± 30	1530 ± 20	2040 ± 10
ω^I (en s^{-1})	-0.9 ± 0.1	-0.8 ± 0.1	-0.37 ± 0.06

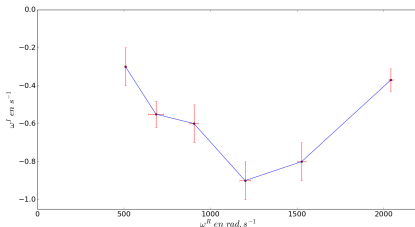


Figure – ω^I en fonction de la corde utilisée

Expérience

Résultats pour les harmoniques d'une seule corde

Pour différentes harmoniques d'une même corde

ω^R (en Hz)	685 ± 6	1360 ± 20	2050 ± 20
ω^I (en s^{-1})	-0.50 ± 0.04	-0.51 ± 0.06	-0.59 ± 0.07
ω^R (en Hz)	2800 ± 20	3450 ± 10	4140 ± 20
ω^I (en s^{-1})	-0.47 ± 0.07	-0.57 ± 0.08	-0.56 ± 0.09

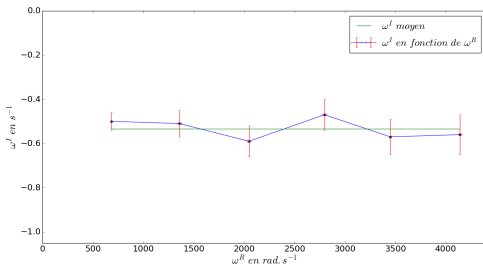


Figure – ω^I en fonction de ω^R pour une même corde

Effet supplémentaire

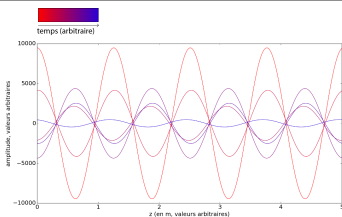


Figure – Variation de la position des noeuds

Sujet étudié par Lukas Novotny en
2007 : Effective Wavelength
Scaling for
Optical Antennas

[http:](http://www2.optics.rochester.edu/workgroups/novotny/papers/novotny07a.pdf)

[//www2.optics.rochester.edu/workgroups/
novotny/papers/novotny07a.pdf](http://www2.optics.rochester.edu/workgroups/novotny/papers/novotny07a.pdf)

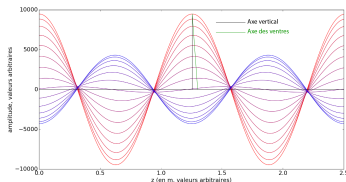


Figure – Variation de la
position des ventres et de la
longueur d'onde

Effet supplémentaire

Conséquence possible

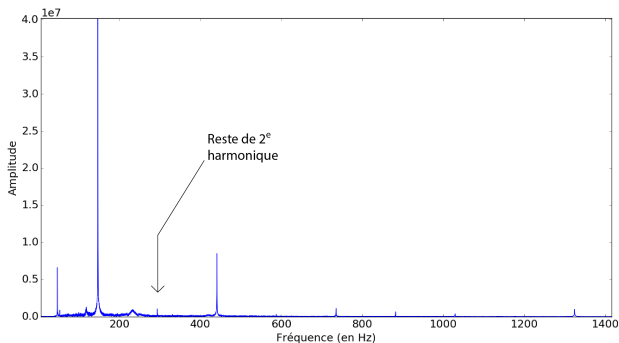


Figure – Harmonique de rang 2 conservée

- 1 Introduction : Corde de Melde
- 2 Cavité semi-ouverte
 - Système
 - Nécessité d'une pulsation complexe
 - Modes quasi-normaux et projection de l'énergie
- 3 Corde Quasi-normale
 - Expression des ondes
 - Expérience
 - Effet supplémentaire
- 4 Conclusion

Conclusion

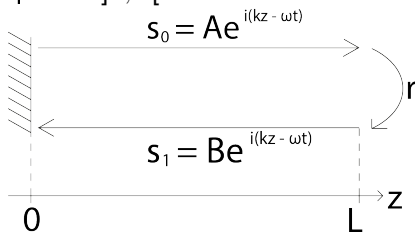
Annexe

- 1 Démonstrations
 - Pulsation complexe
 - Produit scalaire et modes normaux
 - Produit scalaire sur les modes normaux
- 2 Algorithmes utilitaires
 - Modules
 - Maximum et dichotomie
 - Signal centré en 0, dérivation et fondamental
 - Trouver les harmoniques d'un signal
- 3 Algorithmes de traitement numérique
 - Ouverture de fichier
 - Transformée de Fourier
 - Filtre passe-bas
 - Calcul de la pulsation complexe

Pulsation complexe

Pulsation complexe nécessaire

On a le système suivant
avec r coefficient de réflexion tel
que $r \in]0, 1[$



On appelle c la célérité de l'onde
Par continuité en $t = \frac{L}{c}$ on
obtient :

$$rAe^{i(kz - \omega \frac{L}{c})} = Be^{i(kz + \omega \frac{L}{c})}$$

$$\text{D'où } B = rAe^{-2i(\omega \frac{L}{c})}$$

De même en $t = 0$: $A = B$

$$\text{Donc au final } 1 = re^{-2i(\omega \frac{L}{c})} \quad (\star)$$

$$\text{Or } |r| < 1$$

$$\text{Donc } |e^{-2i(\omega \frac{L}{c})}| > 1 \text{ donc}$$

Condition sur ω

$$\omega \in \mathbb{C}$$

Pulsation complexe

Expression de la pulsation

On pose $\omega = \omega^R + i\omega^I$

D'après (★) on obtient

$$\begin{cases} e^{-2i\omega^R \frac{L}{c}} = 1 \\ e^{2\omega^I \frac{L}{c}} = \frac{1}{r} \end{cases}$$

Expression des pulsations

$$\begin{cases} \omega^R = \frac{k\pi c}{L} & k \in \mathbb{Z} \\ \omega^I = \frac{c}{2L} \ln\left(\frac{1}{r}\right) \end{cases}$$

$$\omega = \frac{k\pi c}{L} + i \frac{c}{2L} \ln\left(\frac{1}{r}\right) \quad k \in \mathbb{N}$$

Espace des fonctions à valeurs complexes périodiques

Soit $E = \{f \in \mathcal{C}(\mathbb{R}, \mathbb{C}) \mid f \text{ est } 2\pi\text{-périodique}\}$

Montrons que E est un sous-espace vectoriel de $\mathcal{C}(\mathbb{R}, \mathbb{C})$

$t \mapsto 0$ est 2π -périodique donc $E \neq \emptyset$ Soient $(f, g) \in E$ et $(\alpha, \beta) \in \mathbb{C}$

$$\begin{aligned}\forall t \in \mathbb{R}, (\alpha f + \beta g)(t + 2\pi) &= \alpha f(t + 2\pi) + \beta g(t + 2\pi) \\ &= \alpha f(t) + \beta g(t) \\ &= (\alpha f + \beta g)(t)\end{aligned}$$

Donc $\alpha f + \beta g \in E$

Conclusion

E est un sous-espace vectoriel de $\mathcal{C}(\mathbb{R}, \mathbb{C})$

Produit scalaire

Définition

Définition

Un produit scalaire sur un espace préhilbertien complexe E est une application $\langle . | . \rangle$ vérifiant

- (1)
 $\forall (x, y, z) \in E^3, \forall (\alpha, \beta) \in \mathbb{C}, \langle \alpha x + \beta y | z \rangle = \alpha \langle x | z \rangle + \beta \langle y | z \rangle$
- (2)
 $\forall (x, y, z) \in E^3, \forall (\alpha, \beta) \in \mathbb{C}, \langle x | \alpha y + \beta z \rangle = \bar{\alpha} \langle x | y \rangle + \bar{\beta} \langle x | z \rangle$
- (3) $\forall (x, y) \in E^2, \langle x | y \rangle = \overline{\langle y | x \rangle}$
- (4) $\forall x \in E, \langle x | x \rangle \geq 0$
- (5) $\forall x \in E, \langle x | x \rangle = 0 \implies x = 0_E$

Produit scalaire

Application à E

Soit l'application $\langle .|. \rangle$ définie par

$$\forall (f, g) \in (\mathcal{C}(\mathbb{C}, \mathbb{R}))^2, \langle f|g \rangle = \frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{g(t)} dt$$

Montrons que $\langle .|. \rangle$ est un produit scalaire sur E

Soit $(f, g, h) \in E^3$

Soit $(\alpha, \beta) \in \mathbb{C}^2$

(1)

$$\begin{aligned} \langle \alpha f + \beta g | h \rangle &= \frac{1}{2\pi} \int_0^{2\pi} (\alpha f + \beta g)(t) \overline{h(t)} dt \\ &= \frac{1}{2\pi} \left(\alpha \int_0^{2\pi} f(t) \overline{h(t)} dt \right. \\ &\quad \left. + \beta \int_0^{2\pi} g(t) \overline{h(t)} dt \right) \\ &= \alpha \langle f | h \rangle + \beta \langle g | h \rangle \end{aligned}$$

(2)

$$\begin{aligned} \langle f | \alpha g + \beta h \rangle &= \frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{(\alpha g + \beta h)(t)} dt \\ &= \frac{1}{2\pi} \left(\overline{\alpha} \int_0^{2\pi} f(t) \overline{g(t)} dt \right. \\ &\quad \left. + \overline{\beta} \int_0^{2\pi} f(t) \overline{h(t)} dt \right) \\ &= \overline{\alpha} \langle f | g \rangle + \overline{\beta} \langle f | h \rangle \end{aligned}$$

Produit scalaire

Application à E

(3)

$$\begin{aligned}\langle f|g \rangle &= \frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{g(t)} dt \\ &= \frac{1}{2\pi} \int_0^{2\pi} \overline{\overline{f(t)} g(t)} dt \\ &= \frac{1}{2\pi} \int_0^{2\pi} g(t) \overline{f(t)} dt \\ &= \overline{\langle g|f \rangle}\end{aligned}$$

(4)

$$\begin{aligned}\langle f|f \rangle &= \frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{f(t)} dt \\ &= \frac{1}{2\pi} \int_0^{2\pi} |f(t)|^2 dt \\ &\geq 0\end{aligned}$$

(5) Supposons que $\langle f|f \rangle = 0$

Alors

$$\frac{1}{2\pi} \int_0^{2\pi} f(t) \overline{f(t)} dt = 0$$

Donc

$$\frac{1}{2\pi} \int_0^{2\pi} |f(t)|^2 dt = 0$$

Or $t \mapsto |f(t)|^2$ est continue et positive sur $[0, 2\pi]$

Donc d'après le théorème de positivité

amélioré, $\forall t \in [0, 2\pi], |f(t)| = 0$ Donc

$\forall t \in [0, 2\pi], f(t) = 0$ Or f est

2π -périodique donc $f = 0$

Produit scalaire

Application à E

Conclusion

$\langle . | . \rangle$ est bien un produit scalaire sur E

Application du produit scalaire aux modes normaux

$$\begin{aligned}\forall (s_n, s_p) \in \mathcal{S}^2, \langle s_n | s_p \rangle &= \frac{1}{2\pi} \int_0^{2\pi} e^{in\omega_0 t} e^{-ip\omega_0 t} dt \\ &= \frac{1}{2\pi} \int_0^{2\pi} e^{i(n-p)\omega_0 t} dt \\ &= \begin{cases} \frac{1}{2\pi} \int_0^{2\pi} 1 dt & \text{si } n = p \\ \frac{1}{2\pi} \left[\frac{e^{i(n-p)\omega_0 t}}{i(n-p)\omega_0} \right]_0^{2\pi} & \text{si } n \neq p \end{cases} \\ &= \begin{cases} 0 & \text{si } n = p \\ 1 & \text{si } n \neq p \end{cases}\end{aligned}$$

Conclusion

$\mathcal{S}$ est une famille orthonormale pour $\langle . | . \rangle$

Modules

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 from scipy.io import wavfile
4 import wave
```

Maximum et dichotomie

Trouver un maximum global et son indice

```
1 def self_max(liste):
2     maxi, idx = liste[0], 0
3     for i in range(1, len(liste)):
4         if liste[i] >= maxi:
5             maxi = liste[i]
6             idx = i
7     return maxi, idx
```

Recherche dichotomique

```
1 def dichot(e, elt):
2     ib, it = 0, len(e) - 1
3     while it > ib + 2:
4         med = int((it + ib) / 2)
5         if e[med] < elt:
6             ib = med
7         else:
8             it = med
9     return it
```


Signal centré en 0, dérivation et fondamental

Obtention d'un signal centré en 0

```
1 def offset(e):  
2     moy = 0  
3     for i in range(len(e)):  
4         moy += 1/len(e) * e[i]  
5     return e - moy
```

Dérivation numérique

```
1 def derivative(t, e):  
2     dt, res = t[1] - t[0], np.zeros(len(e))  
3     for i in range(len(e) - 1):  
4         res[i] = (e[i + 1] - e[i])  
5     res[len(e) - 1] = res[len(e) - 2]  
6     return res
```

Trouver le fondamental

```
1 def fundamental(f, s):  
2     i0 = find_peaks(f, s)[0]  
3     return i0, f[i0]
```

Trouver les harmoniques d'un signal

Trouver les pics d'un signal faiblement bruité

```
1 def find_peaks(f, s):
2     amp, s2 = min(s[:100]) - max(s[:100]), np.copy(s)
3     for j in range(len(s2)):
4         if s2[j] < 20 * amp:
5             s2[j] = 0
6     s2prim = derivative(f, s2)
7     res = []
8     for i in range(1, len(s2prim)):
9         if s2prim[i] < 0 and s2prim[i - 1] > 0:
10            res.append(i)
11    #avoid doubles
12    k = 0
13    while k < len(res) - 1:
14        if f[res[k + 1]] - f[res[k]] < 75 or f[res[k]] < 75:
15            res.pop(k + 1)
16        else:
17            k += 1
18    return res
```

Ouverture de fichier wav

```
1 def wav_exploit(fichier):
2     Fe, s = wavfile.read(fichier)
3     #Making the signal
4     signal = np.zeros(len(s))
5     for i in range(len(s)):
6         signal[i] = s[i][1]
7     #Making the time
8     t = np.zeros(len(signal))
9     for i in range(len(signal)):
10        t[i] = i / Fe
11    return Fe, signal, t
```

Transformée de Fourier

Transformée sur tout le signal

```
1 def fourier(t, V):
2     t = np.array(t)
3     Te = t[1] - t[0]
4     Fe = 1/Te
5     V = np.array(V)
6     N = len(V)
7     signal = abs(np.fft.fft(V))
8     frequencies = np.arange(0., N)*Fe/N
9     return frequencies[:N//2], signal[:N//2]
```

Transformée de Fourier

Transformée de fourier sur petits intervalles

```
1 def gliding_fourier(t, e):
2     T0 = t[-1]
3     f, s = fourier(t, e)
4     f0 = fundamental(f, s)[1]
5     T = 10 / f0
6     n_tron = int(T0/T)
7     t_idx = int(len(t) / n_tron)
8     f, s1 = fourier(t[:t_idx], e[:t_idx])
9     res = [s1]
10    time = [0]
11    for i in range(1, n_tron):
12        res.append(fourier(t[i * t_idx:(i + 1) * t_idx], e[i *
13            t_idx:(i + 1) * t_idx])[1])
14        time.append(t[i * t_idx])
15    return f, res, time
```

Filtre passe-bas

D'après la forme canonique $\frac{s}{e} = \frac{H_0}{1+j\frac{\omega}{\omega_0}}$ et en prenant $H_0 = 0$

on obtient $s + s' = e$

qui se discrétise en $s[i+1] = \omega_0 dt \left(e[i] + s[i] \left(\frac{1}{\omega_0 dt} - 1 \right) \right)$

```
1 def low_pass(fe, e, fc = 50):
2     s = np.zeros(len(e))
3     s[0] = e[0]
4     dt = 1/fe
5     w0 = 2 * np.pi * fc
6     for i in range(len(e) - 1):
7         s[i + 1] = (e[i] + s[i] * (1 / (dt * w0) - 1)) * dt * w0
8     return s
```

Calcul de la pulsation complexe

Pour un signal unique

Limites de la régression linéaire

```
1 def limits(e, fe):
2     _, beg = self_max(e)
3     end = beg
4     while end / fe < 0.8 * (len(e) / fe):
5         end += 1
6     return beg, end
```

Calcul des coefficients

```
1 def coef(t, e):
2     fe = 1/(t[1] - t[0])
3     beg, end = limits(e, fe)
4     A, _ = self_max(e)
5     t2 = t[beg:end]
6     e2 = e[beg:end]
7     tmp = np.polyfit(t2, np.log(offset(e2)), 1)
8     return A, tmp[0]
```

Calcul de la pulsation complexe

Pour les différentes harmoniques d'un même signal

```
1 def amplitudes(t, e):
2     x, y = fourier(t,e)
3     peaks = find_peaks(x, y)
4     n = len(peaks)
5     res = [[] for _ in range(n)]
6     f, sps, time = gliding_fourier(t, e)
7     freq = [x[j] for j in peaks]
8     for j in range(n):
9         #j-eme harmonique
10        current_freq = x[peaks[j]]
11        for i in range(len(sps)):
12            #i-eme troncon
13            fj_idx = dichot(f, current_freq)
14            res[j].append(sps[i][fj_idx])
15    return freq, res, time
```