



---

Traitement Numérique des Données  
M1 – INF 2163  
AIDN: Applications Interactives et Données Numériques

# Transformée de Fourier discrète

Sylvie Gibet

1



---

# Séries de Fourier Rappels

2

## Série de Fourier (résumé)

---

Une **fonction périodique**  $s(t)$  qui satisfait les **conditions de Dirichlet** peut être représentée sous la forme d'une série de Fourier, avec des termes reliés d'un point de vue harmonique (termes sinus/cosinus)

## Série de Fourier (rappels)

---

- Soit un **signal périodique** réel :  $s(t) = s(t + kT)$ ,  $T$  période fondamentale
- Alors il existe une **décomposition fondamentale en série de Fourier**

synthèse

$$(1) \quad s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right)$$

## Série de Fourier (rappels)

---

- Avec les  $a_k$  et  $b_k$  : coefficients de Fourier

$$(2) \quad \begin{cases} a_k = \frac{2}{T} \int_{-T/2}^{T/2} s(t) \cos\left(\frac{2\pi kt}{T}\right) dt \\ b_k = \frac{2}{T} \int_{-T/2}^{T/2} s(t) \sin\left(\frac{2\pi kt}{T}\right) dt \\ \frac{a_0}{2} \quad \text{composante continue} \end{cases}$$

analyse

5

## Série de Fourier (Rappels)

---

Avec  $T$  : période du signal =  $1/F$   
 $F$  : fréquence fondamentale  
 $t$  : vecteur temporel  
 $w$  : phase

$$w = 2\pi F$$

6

## Série de Fourier discrète

- On peut transformer les coefficients de la série de Fourier **continue** aux coefficients de la série de Fourier **discrète** en faisant les substitutions suivantes :

$$\square t \rightarrow nT_s \quad T_s : \text{période d'échantillonnage}$$

$$\square \delta t \rightarrow T_s$$

$$\square s(t) \rightarrow s[nT_s] \text{ le signal est échantillonné}$$

7

## Série de Fourier discrète (rappels)

- Soit un **signal périodique** réel :  $s(t) = s(t + kT)$ ,  $T$  période fondamentale
- Alors il existe une **décomposition fondamentale en série de Fourier**

$$(1) \quad s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right)$$

synthèse

avec

$$(2) \quad \begin{cases} a[k] = \sum_{n=0}^{N-1} \cos\left(\frac{2\pi kn}{N}\right) & ; \quad b[k] = \sum_{n=0}^{N-1} \sin\left(\frac{2\pi kn}{N}\right) \\ \frac{a_0}{2} & \text{composante continue} \end{cases}$$

analyse

$a_k$  et  $b_k$  sont les coefficients réels de la décomposition

8

## Série de Fourier discrète (autre écriture)

Signal  $s[n]$ , période =  $N$

**SFD définie par :** SFD engendre des  $c_k$  périodiques de même période que le signal

analyse

$$\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot \left( \cos\left(\frac{2\pi kn}{N}\right) - j \cdot \sin\left(\frac{2\pi kn}{N}\right) \right) = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot e^{-j \frac{2\pi kn}{N}}$$

**Analyse** : détermination des coefficients  $c_k$  (on regroupe  $a_k$  et  $b_k$  avec la notation complexe)

synthèse

$$s[n] = \sum_{k=0}^{N-1} \tilde{c}_k \cdot e^{j \frac{2\pi kn}{N}}$$

$N$  échantillons consécutifs de  $s[n]$  décrivent complètement le signal  $s$  dans les domaines temporel ou fréquentiel.

**Synthèse** : somme finie de sinusoides ( $k$ ) avec coefficients  $c_k$



## Transformée de Fourier

## TF - Un peu d'histoire

---

- Prédictiones astronomiques par Babyloniens/Egyptiens probablement grâce à des sommations trigonométriques
- 1669 : Newton
  - notion de spectre de lumière (*spectre* = fantôme) mais ne découvre pas la notion de "fréquence" (théorie *corpusculaire* & mais pas encore de théorie ondulatoire).

11

## TF - Un peu d'histoire

---

- **18<sup>ème</sup> siècle : deux problèmes non résolus**
  - Les orbites des corps célestes : Lagrange, Euler & Clairaut approximent les données d'observation par des combinaisons linéaires de fonctions périodiques. Clairaut, 1754 (!) première formule de TFD.
  - Les cordes vibrantes: Euler décrit les cordes vibrantes grâce à des sinusoïdes (équation d'ondes). **MAIS** le consensus est que les sommes de sinusoïdes représentent seulement des courbes lisses, donc grosse limitation ...

12

## TF - Un peu d'histoire



- **1807** : Fourier présente son travail sur la conduction de la chaleur  $\Rightarrow$  l'analyse de Fourier est née.
  - $\rightarrow$  Equation de diffusion  $\Leftrightarrow$  série (infinie) de sinus & cosinus. Enorme critique de la part de ses pairs, 1822 (*"Théorie Analytique de la chaleur"*).
- **19<sup>ème</sup> / 20<sup>ème</sup> siècle** : deux voies pour l'analyse de Fourier
  - ▣ Le Continu
  - ▣ Le Discret.

13

## TF - Un peu d'histoire



- **Le CONTINU**
  - ▣ Fourier étend l'analyse à des fonctions arbitraires (Transformation de Fourier).
  - ▣ Dirichlet, Poisson, Riemann, Lebesgue attaquent la convergence des SF.
  - ▣ D'autres variantes naissent en fonction des besoins (ex.: TF à court terme – analyse de la parole).
- **Le DISCRET : méthode de calcul rapide (FFT: Fast Fourier Transform)**
  - ▣ **1805** - Gauss, 1<sup>ère</sup> utilisation de la FFT (manuscrit en Latin oublié!!! Publié en 1866).
  - ▣ **1965** - IBM Cooley & Tukey "redécouvrent" l'algorithme de la FFT (*"An algorithm for the machine calculation of complex Fourier series"*).
  - ▣ raffinement de l'algorithme de FFT pour la plupart des ordinateurs.

14

## Transformée de Fourier

- Il existe plusieurs termes pour définir la transformée de signaux continus et discrets
  - **Séries de Fourier (SF)** : s'applique aux signaux **continus périodiques** – produit un spectre discret
  - **Transformée de Fourier (TF)** : s'applique aux signaux continus apériodiques – produit un spectre continu apériodique
  - **Transformée de Fourier à temps discret (TFTD)** : s'applique aux signaux discrets apériodiques – produit un spectre continu périodique
  - **Transformée de Fourier discrète (TFD)** : s'applique aux signaux discrets apériodiques – produit un spectre discret périodique
  - **Séries de Fourier discrète (SFD)** : sert d'approximation aux coefficients de la SF

15

## Transformée de Fourier discrète

- La transformée de Fourier discrète est une méthode qui permet de décrire un signal discret en fonction de la fréquence
  - ▣ Applicable aux signaux discrétisés dans le temps ou dans l'espace (exemple du son et de l'image)
  - ▣ Même forme que la SFD mais applicable aux signaux apériodiques
  - ▣ **La seule transformée qui s'applique aux problèmes de traitement numérique des données est la TFD : transformée de Fourier Discrète**

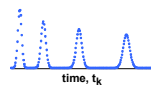
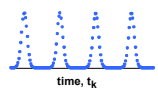
16



## Transformée de Fourier discrète

Signal temporel d'entrée

Spectre fréquentiel



Discret

Périodique (période  $T$ )   
 SFD\*\* Discret   
 Apériodique {   
 TFTD Continu   
 TFD\*\* Discret

$$\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot e^{-j \frac{2\pi k n}{N}}$$

$$S(f) = \sum_{n=-\infty}^{+\infty} s[n] \cdot e^{-j 2\pi f n}$$

$$\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot e^{-j \frac{2\pi k n}{N}}$$

\*\* Calculée via FFT

Note:  $j = \sqrt{-1}$ ,  $\omega = 2\pi/T$ ,  $s[n] = s(t_n)$ ,  $N = \text{Nombre d'échantillons}$

## Transformée de Fourier Discrète

TFD définie par :

analyse

$$\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot e^{-j \frac{2\pi k n}{N}}$$

synthèse

$$s[n] = \sum_{k=0}^{N-1} \tilde{c}_k \cdot e^{j \frac{2\pi k n}{N}}$$

Résolution fréquentielle

Fréquences d'analyse  $f_k$

$$f_k = \frac{k \cdot f_s}{N}, k = 0, 1, 2, \dots, N-1$$



Les échantillons de la TFD sont localisés sur les fréquences d'analyse  $f_k$

Note :  $c_{k+N} = c_k \Leftrightarrow$  Le spectre est périodique, de période  $N$

## Transformée de Fourier Discrète – exemple

Calculer la TFD du signal suivant :  $x[n] = \{1, 2, 1, 0\}$

- $N = 4$ , et donc  $e^{-j\frac{2\pi kn}{N}} = e^{-j\frac{\pi kn}{2}}$
- $k=0$   $\tilde{c}_0 = \sum_{n=0}^3 x[n] \cdot e^0 = 1 + 2 + 1 + 0 = 4$
- $k=1$   $\tilde{c}_1 = \sum_{n=0}^3 x[n] \cdot e^{-j\frac{\pi n}{2}} = 1 + 2e^{-j\frac{\pi}{2}} + e^{-j\pi} + 0 = -j2$
- $k=2$   $\tilde{c}_2 = \sum_{n=0}^3 x[n] \cdot e^{-j\pi n} = 1 + 2e^{-j\pi} + e^{-j2\pi} + 0 = 0$
- $k=3$   $\tilde{c}_3 = \sum_{n=0}^3 x[n] \cdot e^{-j\frac{3\pi n}{2}} = 1 + 2e^{-j\frac{3\pi}{2}} + e^{-j3\pi} + 0 = j2$

Et donc  $X_{\text{TFD}} = \{4, -j2, 0, j2\}$

19

## TFD - exemples

Quelques transformées de Fourier utiles

- $\{1, 0, 0, 0, \dots, 0\}$  impulsion
- $\{1, 1, 1, \dots, 1\}$  constante
- $\cos(2\pi n k_0 / N)$  sinusoïde
- $\{1, 1, 1, 1, \dots, 1\}$  constante
- $\{N, 0, 0, 0, \dots, 0\}$  impulsion
- $0.5N(\delta[k - k_0] + \delta[k - (N - k_0)])$

20

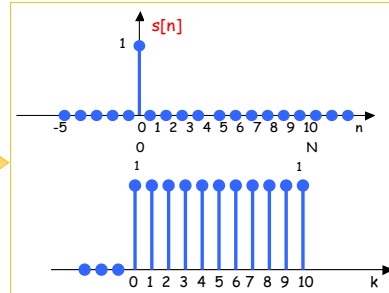
## TFD - exemples

### Impulsion, largeur N, discret

$$\delta[n] = \begin{cases} 1, & \text{si } n=0 \\ 0, & \text{ailleurs} \end{cases}$$

$$\tilde{c}_k = \frac{1}{N} \sum_{n=0}^{N-1} s[n] \cdot e^{-j\frac{2\pi kn}{N}} = \frac{1}{N} e^{-j\frac{2\pi kn}{N}}$$

$$\forall k, \tilde{c}_k = \frac{1}{N}$$

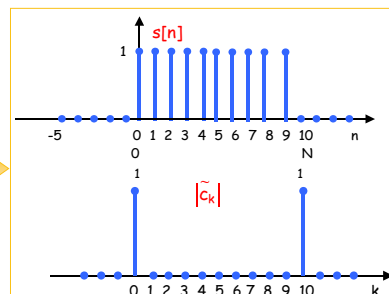


## TFD - exemples

### Signal rectangulaire, largeur N, discret

$$u[n] = \begin{cases} 1, & \text{if } 0 \leq n \leq N-1 \\ 0, & \text{sinon} \end{cases}$$

$$\tilde{c}_k = (1/N) e^{-j\pi k(N-1)/N} \sin(\pi k) / \sin(\pi k/N)$$



Remarque : un signal qui est court dans le temps est long en fréquence et vice-versa

## TFD - exemples

**Sinusoïde, fréquence  $f_0$**

$$s[n] = \cos(2\pi k_0 n/N)$$

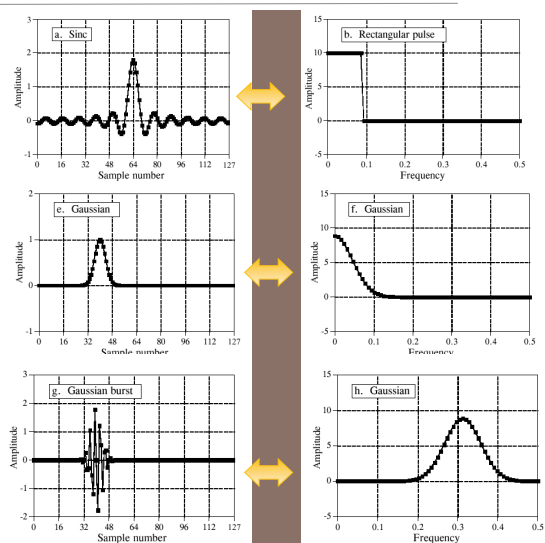
$N$  échantillons

$$c_k = (1/N) e^{j\pi\{(Nf_0-k)-(Nf_0-k)/N\}} (1/2) \sin\{\pi(Nf_0-k)\} / \sin\{\pi(Nf_0-k)/N\} + \\ (1/N) e^{j\pi\{(Nf_0+k)-(Nf_0+k)/N\}} (1/2) \sin\{\pi(Nf_0+k)\} / \sin\{\pi(Nf_0+k)/N\}$$

soit une valeur (amplitude) de  $0.5 N (\delta[k - k_0] + \delta[k - (N - k_0)])$

## TFD – Autres exemples

Les TFD correspondent aux versions échantillonnées de la TFD avec fenêtrage



## Fast Fourier Transform

---

- FFT

$X = \text{fft}(x)$  (X est complexe !)

- On peut utiliser 2 formes de représentation de la TFD

- ▣ forme réelle et imaginaire (complexe)
- ▣ forme polaire (amplitude et phase)

- Convention : on utilise une minuscule pour représenter un signal en fonction du temps ( $x[n]$ ) et une majuscule pour représenter un signal en fonction de la fréquence ( $X[n]$ )

25

## Fast Fourier Transform (FFT)

---

- Il existe plusieurs méthodes pour calculer la TFD, mais la plus rapide est la FFT (Fast Fourier Transform)
- J.W. Cooley et J.W. Tukey : 2 ingénieurs à l'origine de l'invention de la FFT (article paru en 1965)
- TFD : opération d'ordre  $N^2$  multiplications
- FFT :  $\log_2(N)$  multiplications → beaucoup plus rapide !

## FFT et Python

### □ Forme réelle et imaginaire

$$X(f) = \text{real}(X(f)) + j \text{imag}(X(f))$$

### □ Forme polaire

$$X(f) = \text{abs}(X) \cdot \exp(j \Phi(f))$$

$\text{abs}(X(f))$  : amplitude du spectre (magnitude)

$\Phi(f) = \text{angle}(X(f))$  : phase du spectre

27

## FFT et Python

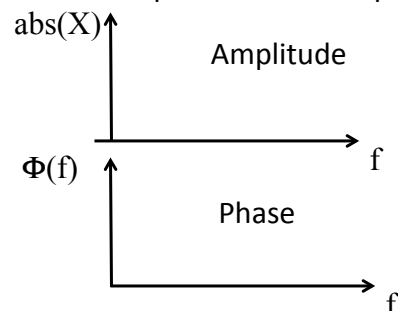
### □ En Python : $X = \text{fft}(x)$

$X$  : transformée de Fourier du signal  $x$  échantillonné

Représentation temporelle



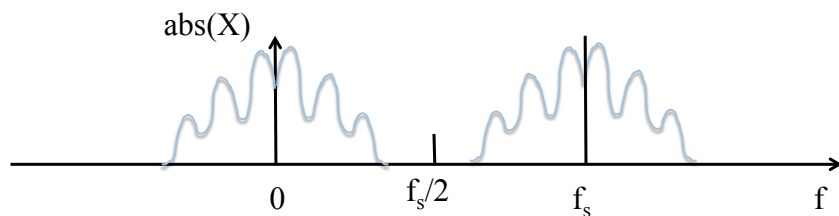
Représentation fréquentielle



28

## Transformée de Fourier discrète

Représentation fréquentielle : périodicité



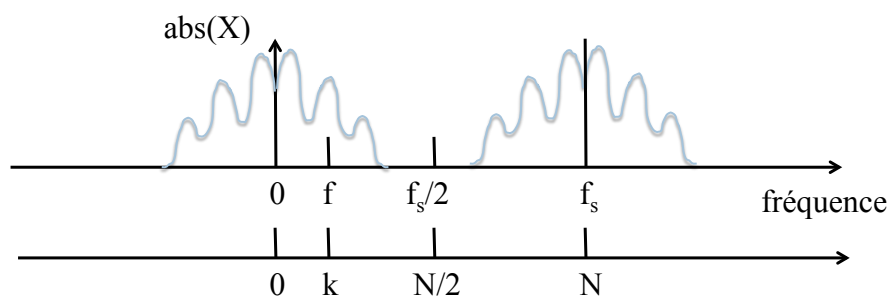
Nécessité d'avoir  $f_s \gg f_{\max}$  (théorème de Shannon-Nyquist)

Sinon, phénomène de repliement de spectre ! (voir en TP)

29

## Transformée de Fourier discrète

Échelle fréquentielle discrète



$$TF\{x[n]\} = \{X[k]\} \quad \text{avec } k/N = f/f_s$$

$x[n]$  échantillonné : 0 à N échantillons (échelle temporelle)

$X[k]$  échantillonné : 0 à  $f_s$  échantillons, avec  $k/N = f/f_s$

30

## Propriétés de la TFD

■ Si  $X_{\text{TFD}}$  est la TFD d'un signal discret  $x[n]$ , alors on a :

$kx[n]$	$\longleftrightarrow$	$kX_{\text{TFD}}[k]$	<b>linéarité</b>
$x_1[n] + x_2[n]$	$\longleftrightarrow$	$X_1[k] + X_2[k]$	<b>additivité</b>
$x[-n]$	$\longleftrightarrow$	$X_{\text{TFD}}[-k] = X_{\text{TFD}}^*[k]$	<b>repliement</b>
$x[n-s]$	$\longleftrightarrow$	$e^{-j\frac{2\pi sk}{N}} X_{\text{TFD}}[k]$	<b>déphasage</b>
$x[n].y[n]$	$\longleftrightarrow$	$1/N X_{\text{TFD}}[k] * Y_{\text{TFD}}[k]$	<b>multiplication</b>
$x[n]*y[n]$	$\longleftrightarrow$	$1/N X_{\text{TFD}}[k].Y_{\text{TFD}}[k]$	<b>convolution</b>

\* $X^j$  :  $X$  conjugué



## Exemples

### Analyse spectrale d'un signal



## Exemple 0 – code Python

### □ Echelle temporelle

$x[n]$  échantillonné : 0 à N échantillons (échelle temporelle)

```
n = np.arange(N) #création d'une échelle temporelle de
```

```
t = n / fs          # 0 à N-1
```

ou

```
t = np.linspace(0,(N-1)/fs, N)
```

### □ Echelle fréquentielle

$X[k]$  échantillonné : 0 à  $f_s$  échantillons, avec  $k/N = f/f_s$

```
F = np.linspace(0,fs,N), avec N puissance de 2
```

33

## Exemple 0

- Création d'un signal échantillonné (4 sinusôides aux fréquences 1, 2, 3 et 4 Hz)

```
def genSin (N,f,fs) :
```

```
    t = np.linspace(0,(N-1)/fs, N) ; pi = np.pi
```

```
    x=np.sin(2*pi*f*t);
```

```
    return(x)
```

```
fe = 10 ; N=1024 ; T = float(N-1)/fe
```

```
echantillons = np.zeros(N)
```

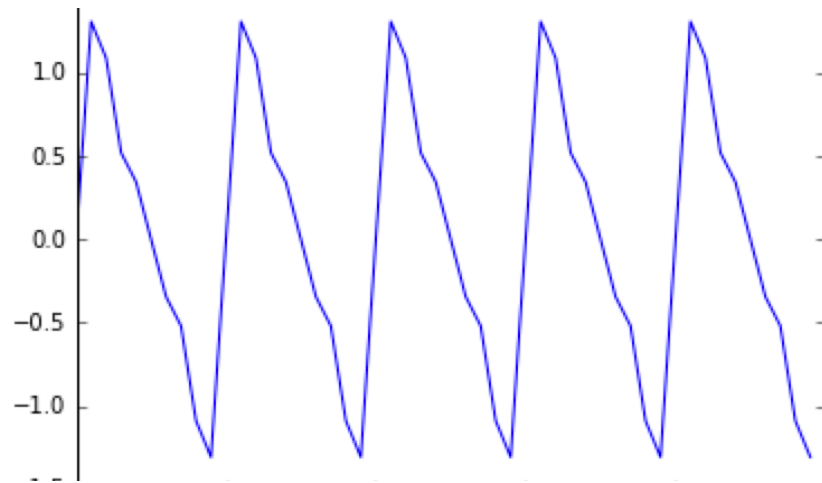
```
echantillons = genSin(N,1,fe)+0.5*genSin(N,2,fe)+0.25*genSin(N,3,fe)  
+0.01*genSin(N,4,fe)
```

```
n=np.arange(N)
```

```
plt.plot(n[0:50], echantillons[0:50])
```

34

## Exemple 0 : signal temporel synthétisé



35

## Exemple 0 - Spectre du signal

□ Spectre du signal – parties réelle et imaginaire

`TFD = fft(echantillons)` # spectre du signal

`Re = np.real(TFD)` # partie réelle

`Im = np.imag(TFD)` # partie imaginaire

36

## Exemple 0 – Affichage du spectre

### □ Partie réelle du spectre

```
freq = np.linspace(0,fe,N)
```

```
figure(figsize=(10,4))
```

```
plot(freq,Re,'g')
```

```
xlabel('fréquence')
```

```
ylabel('Réelle')
```

```
axis([0,fe/2,0,Re.max()])
```

```
grid()
```

### □ Partie imaginaire

```
figure(figsize=(10,4))
```

```
plot(freq,Im,'g')
```

```
xlabel('fréquence')
```

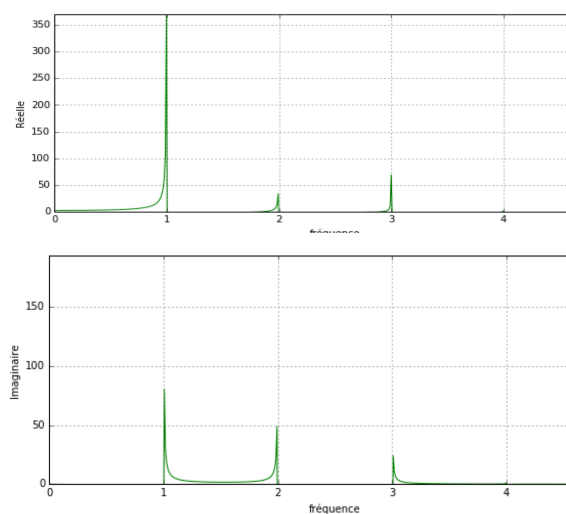
```
ylabel('Imaginaire')
```

```
axis([0,fe/2,0,Im.max()])
```

```
grid()
```

37

## Spectre du signal : parties réelles et imaginaires



38

## Spectre du signal - calcul

- Spectre du signal : amplitude (magnitude et phase)

```
TFD = fft(echantillons)    # spectre du signal
A = np.absolute(TFD/N)      # amplitude du spectre
An = A/A.max()              # amplitude normalisée
P = np.angle(TFD/N)         # phase normalisée
```

39

## Spectre du signal - affichage

- Amplitude

```
freq = np.linspace(0,fe,N)
```

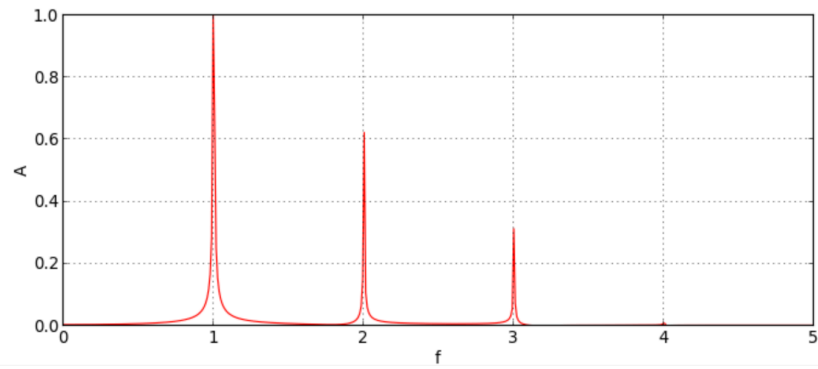
```
figure(figsize=(10,4))
plot(freq,An,'b')
xlabel('fréquence')
ylabel('Amplitude')
axis([0,fe/2,0,An.max()])
grid()
```

- Phase

```
figure(figsize=(10,4))
plot(freq,P,'r')
xlabel('fréquence')
ylabel('Phase')
axis([0,fe/2,0,P.max()])
grid()
```

40

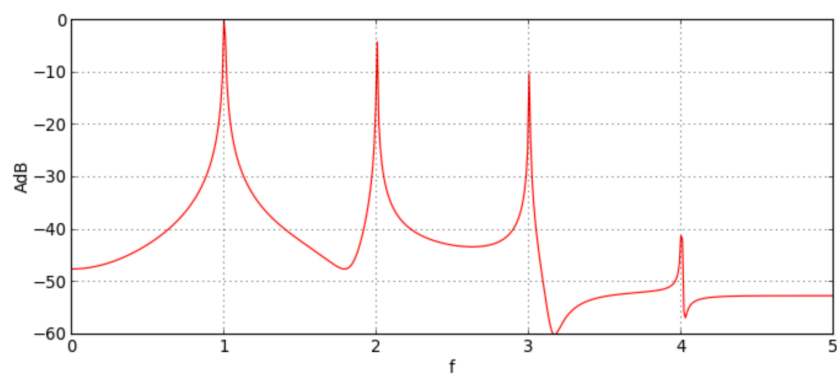
### Spectre du signal : amplitude



41

### Spectre du signal : en dB

$\text{spectre\_db} = 20 * \text{np.log10}(A_n)$



42

## Spectre du signal : en dB

---

- Si le signal était échantillonné sur une durée infinie, les raies seraient de largeur nulle (puisque le signal est périodique).
- L'élargissement de la base des raies est un effet de la durée finie de l'échantillon, c'est-à-dire de l'application d'une fenêtre de troncature au signal.
- S'il n'est pas gênant dans le cas d'un spectre discret, cet élargissement peut réduire la résolution dans le cas d'un spectre continu.
- On constate également des erreurs sur les hauteurs relatives des raies, qui viennent du fait que la résolution fréquentielle (inverse de la durée de l'échantillon) est insuffisante pour saisir le maximum des raies.

43

## Exemple 1

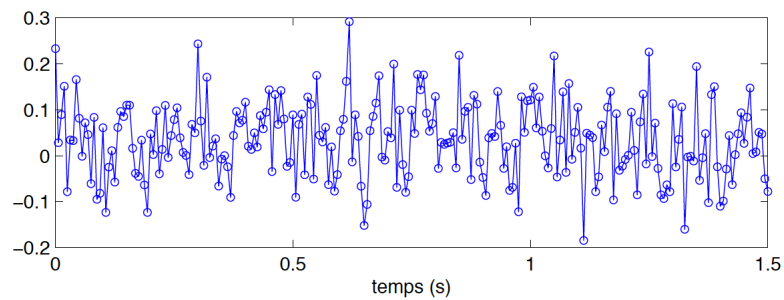
---

- Supposons qu'on veuille caractériser les sons enregistrés dans l'océan.
- On place un microphone dans l'eau pour enregistrer les sons, puis on amplifie le signal.
- On utilise un filtre passe-bas pour couper les sons au-dessus de 80Hz.
- On échantillonne à 160Hz.
- On procède ensuite à l'analyse du spectre.

44

## Exemple 1

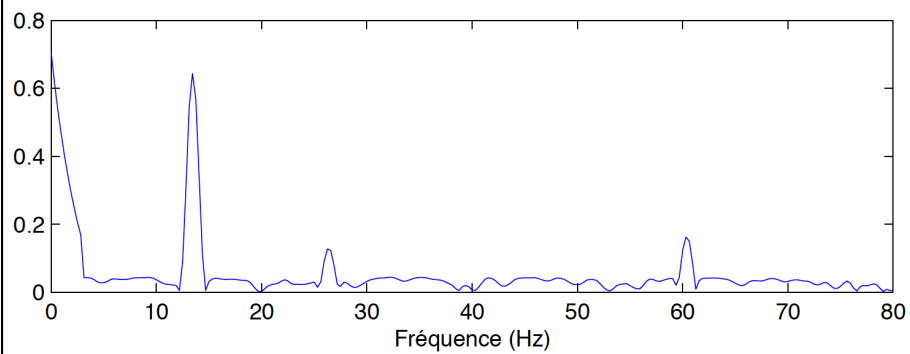
- On observe le signal en fonction du temps pour voir si on peut déduire de l'information.



45

## Exemple 1

Si l'on observe le spectre :



46

## Exemple 1

---

### Analyse du contenu spectral

- Si on ignore les pics, on voit qu'entre 10Hz et 70Hz le spectre est constant ; c'est le bruit blanc gaussien.
- Le bruit blanc gaussien est appelé ainsi parce qu'il est constant sur toutes les fréquences.
- Le bruit blanc est causé par plusieurs choses : ici, ça peut être le microphone, ou même l'océan.

47

## Exemple 1

---

### Analyse du contenu spectral (suite)

- Pour les fréquences très faibles, on remarque que le bruit augmente très rapidement, avec une forme qui ressemble à  $1/f$  : c'est le bruit rose (flicker noise).
- Ce type de bruit apparaît dans pratiquement tous les systèmes physiques, que ce soit électrique, mécanique, etc. Aucune théorie ne permet d'expliquer tous les cas où se trouve ce type de bruit.
- Pour les systèmes électriques courants, c'est généralement en-dessous de 100Hz qu'on retrouve ce type de bruit.

48



## Exemple 1

---

### Analyse du contenu spectral (suite) : les pics

- Il y a un pic important à 13Hz, et un pic plus faible à 26Hz, qui est probablement une harmonique de celui de 13Hz.
- Ce signal pourrait être causé par les pâles triples d'un sous-marin qui tournent à 4.33 tours/seconde.
- Cette technique est la base du sonar passif.

49

## Exemple 2

---

### Obtention du spectre à partir des échantillons d'un signal

- Pour obtenir le spectre d'un signal à l'aide de la TFD, il suffit d'appliquer l'équation de la TFD aux échantillons.
- Soit le signal :

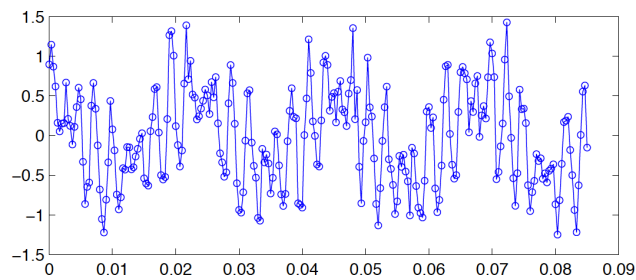
$$x(t) = 0.5 \cos(2\pi 44t) + 0.4 \sin(2\pi 232t) + 0.5 \cos(2\pi 415t)$$

- On va ajouter du bruit blanc gaussien à ce signal
- Le signal est échantillonné à 3kHz, et on utilise 256 échantillons.

50

## Exemple 2

Le signal est le suivant :

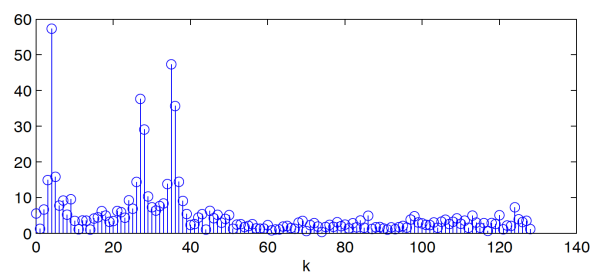


$N = 256$  ;  $F_s = 3000$   
 $t = \text{np.linspace}(0, (N-1)/F_s, N)$   
 $x = 0.5 * \cos(2 * \pi * t * 44) + 0.4 * \sin(2 * \pi * t * 322) + 0.5 * \cos(2 * \pi * t * 415)$

51

## Exemple 2

A l'aide de la TFD, on obtient le spectre suivant :



$X = \text{fft}(x)$

52

## Exemple 2

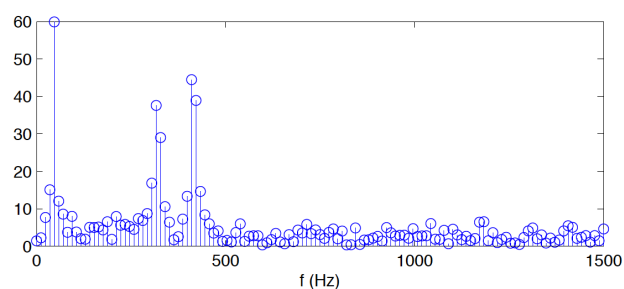
Quelques remarques sur la figure précédente :

- L'abscisse est en fonction de  $k$ , et pas en fonction de la fréquence.
- Il faut transformer l'abscisse de sorte qu'elle soit en fonction de la fréquence d'échantillonnage, si on veut que le graphe ait du sens
- Les points de 0 à 128 doivent varier de 0 à 1500Hz (d'après le théorème de Shannon Nyquist).

53

## Exemple 2

En fonction de la fréquence :

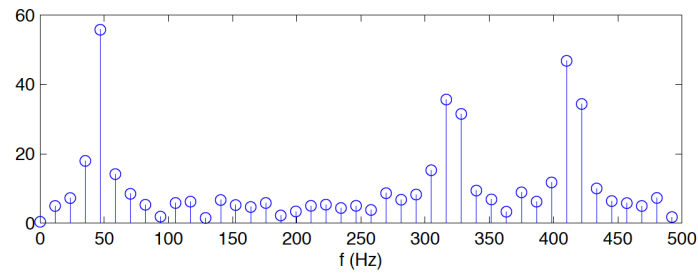


```
f = np.linspace(0,Fs/2,N/2)
plt.stem(f[0:128],abs(X[0:128]))
```

54

## Exemple 2

Comme il y a peu d'information utile après 500Hz, on fait un zoom sur cette partie du spectre à l'aide de la commande *xlim*.

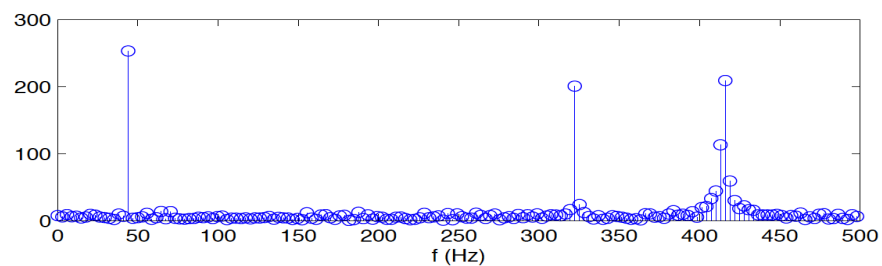


Cependant, la résolution spectrale ( $F_s/N$ ) est de 11.72Hz, ce qui ne permet pas de bien déterminer les fréquences. Il faudrait utiliser plus de points.

55

## Exemple 2

Avec 1024 points, la résolution est bien meilleure.



Cependant, plus on a de points, plus le temps de calcul est élevé.

56

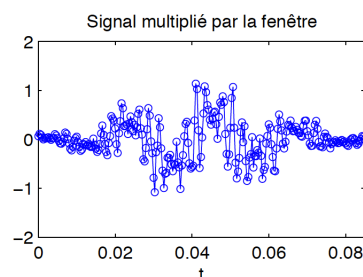
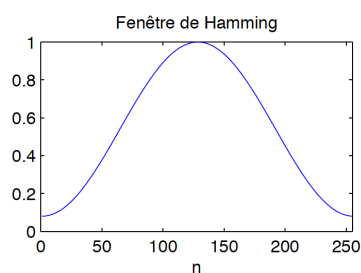
## Fenêtre de Hamming

- Une façon d'améliorer les pics sans utiliser plus de points est de multiplier le signal en fonction du temps par une composante appelée fenêtre de *Hamming*.
- La fenêtre de *Hamming* va atténuer les bouts du signal. En terme de spectre, la fenêtre de *Hamming* va rendre les pics plus grands par rapport au bruit, mais ça va aussi les élargir.
- Il existe plusieurs autres fenêtres.

57

## Fenêtre de Hamming

- Effet de la fenêtre de *Hamming* :



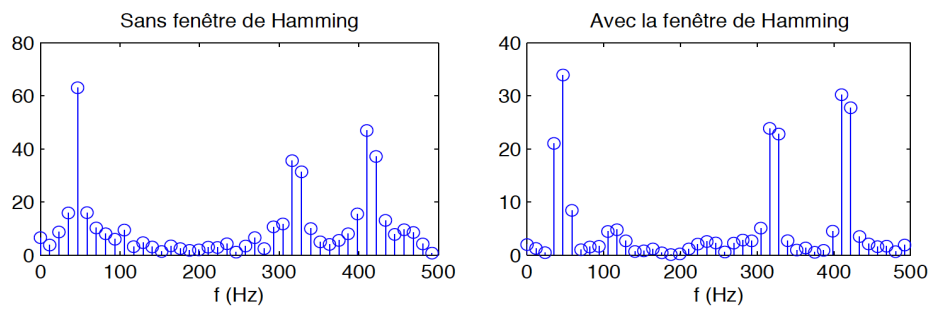
- Définie sur la fenêtre  $[0, T]$  par :

$$w(t) = 0.54 - 0.4 \cos(2\pi/T \cdot t)$$

58

## Fenêtre de Hamming

- Effet de la fenêtre de *Hamming* sur le spectre



59

## Applications

- Analyse spectrale d'un signal sonore

## Analyse spectrale du son

---

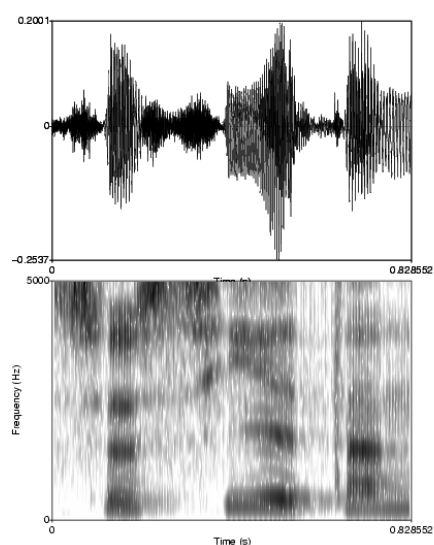
- SIGNAL : représentation en temps / amplitude
- SPECTRE : représentation en fréquence / amplitude
- SPECTROGRAMME : représentation en temps / fréquence

## Analyse spectrale du son

---

- Signal de parole

Signal



Spectrogramme

## Sons harmoniques

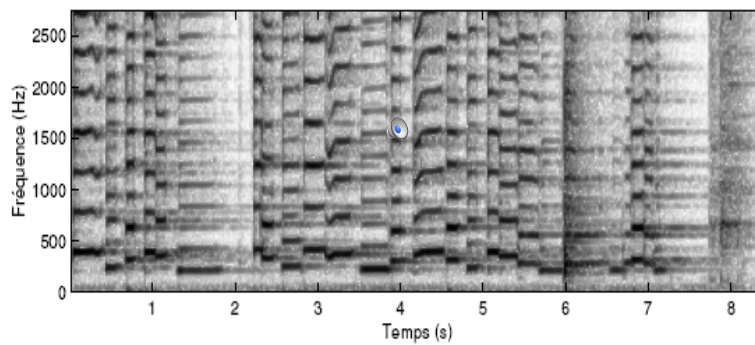


FIG. 1.4 – Sonogramme d'un son instrumental (saxophone de Sylvain Beuf), Piste n° 3 🎵.

## Sons bruités

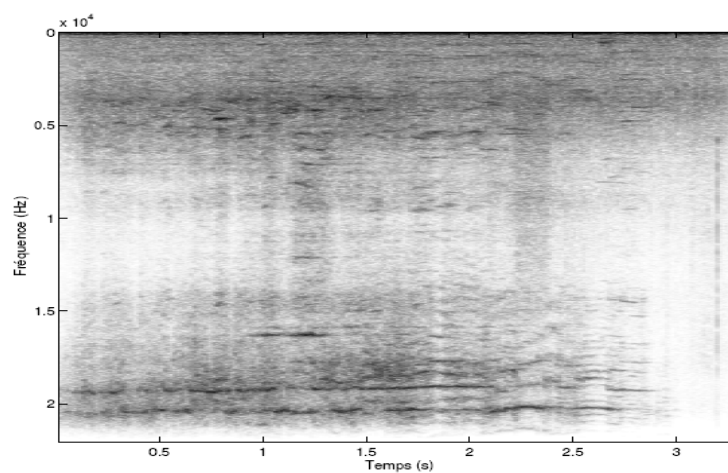


FIG. 1.5 – Sonogramme d'un bruit de freins de bus, Piste n° 56-CD2 🎵.



## Sons quasi-harmoniques, inharmoniques

### □ Entre les 2, on a :

- des sons quasi-harmoniques : les partiels ont leur fréquence qui dévie petit à petit du multiple de la fréq fond. (cas de sons instrumentaux : marimba, xylophone)
- sons inharmoniques : les partiels ne sont pas multiples d'une fondamentale ; cependant on peut entendre une hauteur, voire plusieurs (sons de cloche)
- sons filtrés : il s'agit de bruits filtrés passe-bande étroit. Selon la structure du filtre, une hauteur peut apparaître lors de leur audition (filtre résonant).

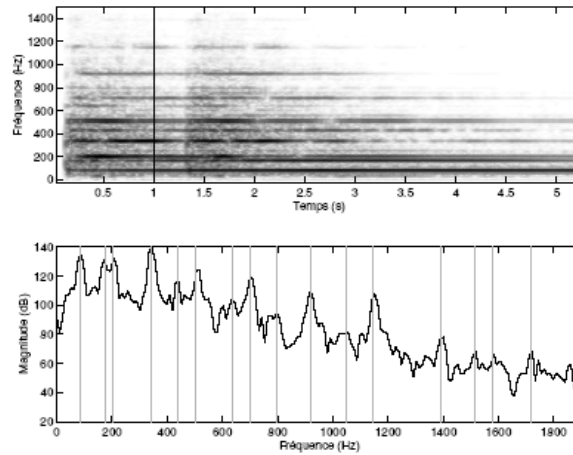
65

## Enveloppe spectrale

- **Fréquence fondamentale** : Un son parfaitement périodique se décompose comme la somme de sinusoides de fréquences  $f_n = nf_0$  avec  $f_0 = f_1$ .  
 $f_0$  est la fréquence fondamentale,  $n$  est le numéro d'harmonique
- **Enveloppe spectrale** : si l'on observe le spectre d'amplitude des signaux musicaux, en regardant leur évolution dans le temps, on remarque que des formes plus générales que les partiels apparaissent : il s'agit de l'**enveloppe spectrale**, définie comme l'enveloppe supérieure (forme reliant les pics ou partiels), souvent approchée par lissage du spectre.  
Enveloppe formée de bosses : **formants** et de creux.

66

## Sons quasi-harmoniques, inharmoniques



*Spectrogramme (fig. bas) d'un son de cloche, Piste n° 2 🎵 et sa TCFT (fig. bas) prélevée au niveau de la ligne verticale noire, à 1 s. Les traits gris verticaux sur la TFCT indiquent la fréquence de chaque partiel.*

## Harmoniques, formants

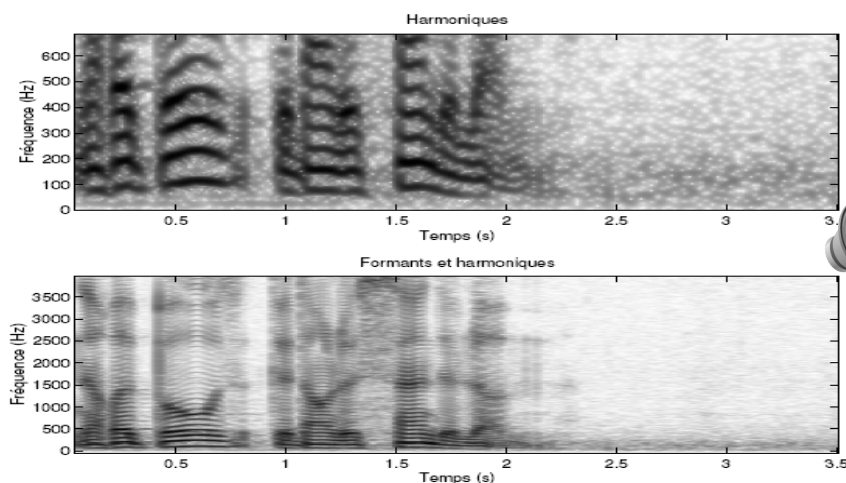


FIG. 1.7 – Zoom sur le sonagramme d'une voix parlée : fréquences de 0 à 700 Hz pour visualiser les harmoniques, et de 0 à 4000 Hz pour visualiser les formants, pour le son Piste n° 16 🎵.

## Analyses de Fourier - Applications

---

L'analyse de Fourier a révolutionné les mathématiques : la transformée de Fourier permet de retrouver les fréquences distinctes d'un même signal

- **Télécommunications** – téléphonie GSM/cellulaire, ADSL: *Asymmetric Digital Subscriber Line. (raccordement numérique asymétrique, RNA ou liaison numérique à débit asymétrique)*
- **Musique, audio, multimedia** : compression MP3
- **Electronique/IT** – la plupart des applications basées Digital Signal Processing (DSP)
- **Traitement d'image** : compression, imagerie médicale

## Analyses de Fourier - Applications

---

**Théorie des ondelettes** : *Obtenir des informations de fréquences et de temps dans un signal (inventé par un ingénieur de chez Elf-Aquitaine)*

- Sorte de microscope mathématique permettant de zoomer sur un instant du signal, et d'accéder à des détails très fins
- Avec la TF, l'information temporelle est perdue -> adaptation de la TF en l'appliquant à différents morceaux du signal, les fenêtres temporelles étant d'autant plus courtes qu'on accède aux fréquences élevées.
- Intéressant pour détecter des changements brusques dans un signal.

Applications : format de compression d'image JPEG 2000

## Analyses de Fourier - Applications

---

- *Analyse de la parole* (reconnaissance, synthèse, etc.)  
*Industrie/recherche* - X-ray spectrometry, analyse chimique (FT spectrometry), conception de radar, d'antennes
- *Médecine*
  - PET (Positron Emission Tomography)-scanner : technique d'imagerie nucléaire
  - IRM : Imagerie par Résonance Magnétique
- Etc.



## Exercices

## Fréquence de Nyquist/Shannon

---

- A quelle fréquence doit on échantillonner le signal suivant ?

$$0.5 + 0.5.\cos(1600.\pi.t) + \sin(1200.\pi.t) - 2.\cos(400.\pi.t)$$

- Le signal précédent est-il décomposable en série de Fourier ?  
Pourquoi ?

## Périodicité, séries de Fourier

---

- Les fonctions suivantes sont-elle périodiques, et si oui, quelle est la période ?

$$\cos(t) - \sin(2.t)$$

$$5.\cos(3t) - 2.\sin(6.t)$$

$$\cos(t) + \cos(\pi.t)$$

## Séries de Fourier

---

- Donnez les coefficients de Fourier des signaux suivants :

$$0.5 + 0.5.\cos(1600.\pi.t) + \sin(1200.\pi.t) - 2.\cos(400.\pi.t)$$

$$\cos(t) - \sin(2.t)$$

$$5.\cos(3t) - 2.\sin(6.t)$$



## Compléments du cours

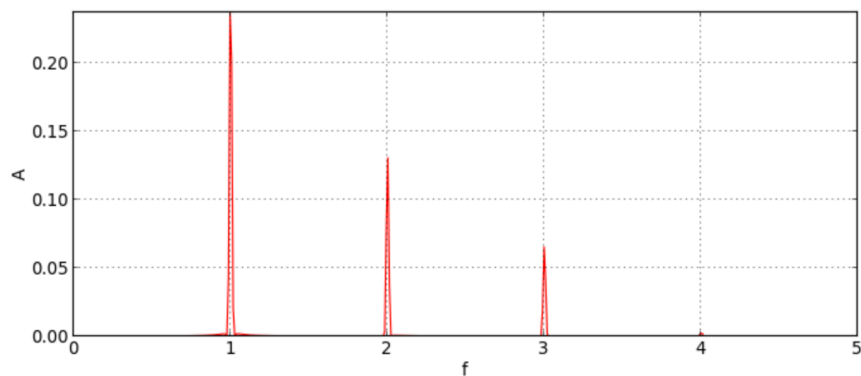
## Application d'une fenêtre de Hamming

```

hamming = signal.hamming(N)
figure(figsize=(8,4))
plot(hamming)
xlabel('k')
ylabel('A')
grid()
echantillons = echantillons*hamming
spectre = np.absolute(fft(echantillons))/N
figure(figsize=(10,4))
plot(freq,spectre,'r')
xlabel('f')
ylabel('A')
axis([0,fe/2,0,spectre.max()])
grid()
    
```

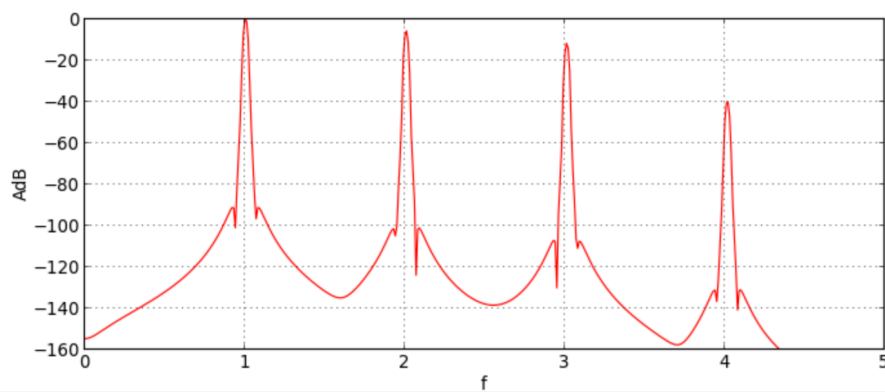
77

## TFD du signal avec fenêtre de Hamming



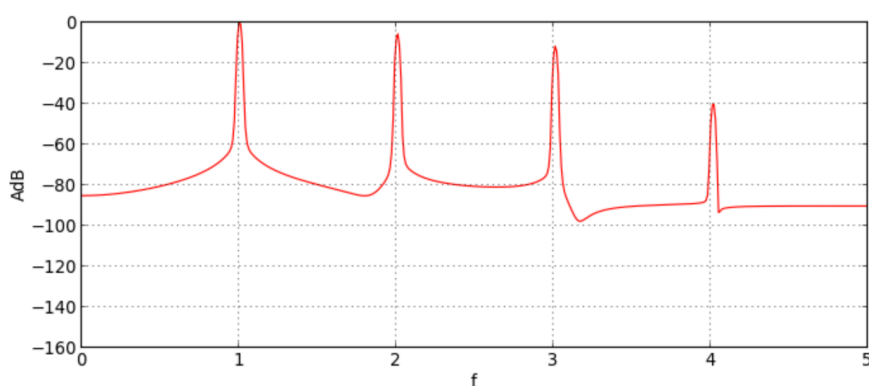
78

### TFD du signal avec fenêtrage de Hamming (en dB)



79

### TFD du signal avec fenêtrage de Hanning (en dB)



80