J.COURTIN — ***PSI*** — *LYCÉE V.HUGO*

Électronique Numérique

Objectifs :

- Échantillonnage
- Condition de Nyquist-Shannon
- Analyse spectrale & filtrage numériques

Pré-requis :

- Numérisation d'un signal [info représentation binaire]
- TP Échantillonnage
- Décomposition harmonique

Pourquoi numériser un signal ?




Le traitement directe de données analogiques issues de capteurs posent un certain nombre de difficultés :

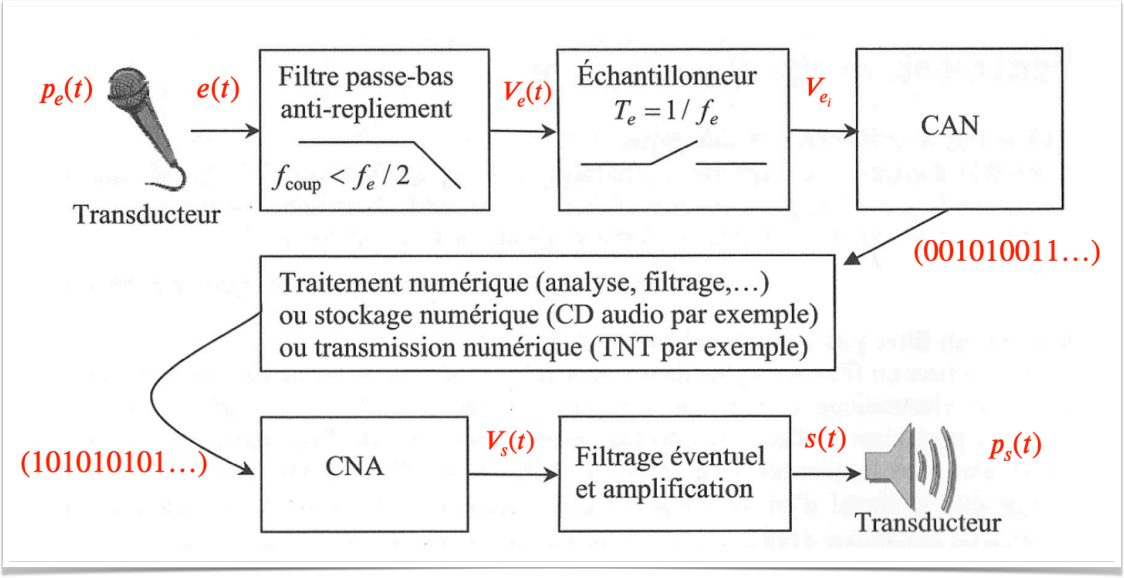
- Les opérations réalisables sont limitées [l'électronique est plus avancée que le reste : amplification, déphasage, filtrage, dérivation, intégration, sommation, soustraction, multiplication]
- Elles sont généralement irréversibles et sensibles aux perturbations lors du transfert de données, donc peu fiables.
- Elles sont généralement complexes à mettre en oeuvre (coût des dispositifs, encombrement) ⇒ coût énergétique.
- Elles ne sont pas particulièrement rapides et difficile à conserver dans le temps.
- Enfin le stockage des données analogiques recouvrent également tous ces problèmes [disque vinyle, photographie, bande magnétique]

La numérisation des signaux : implique au départ une perte des données [contrôlable], mais la gestion informatique de ces données est à la fois rapide et efficace car les mémoires de stockage et les unités de calcul sont largement compatibles entre elles, de taille réduite, de faible encombrement ⇒ système embarqué et traitement in situ.

Notons toutefois que tout capteur est analogique, mais l'électronique est tellement miniaturisée que l'on peut aujourd'hui intégré la conversion numérique au sein même du capteur [accéléromètre, caméra, thermocouple] On peut ainsi envoyer toutes ces données vers un calculateur central qui gère l'ensemble des données numérisées car **les données sont toutes sous la même forme**, le transfert rapide et fiable.

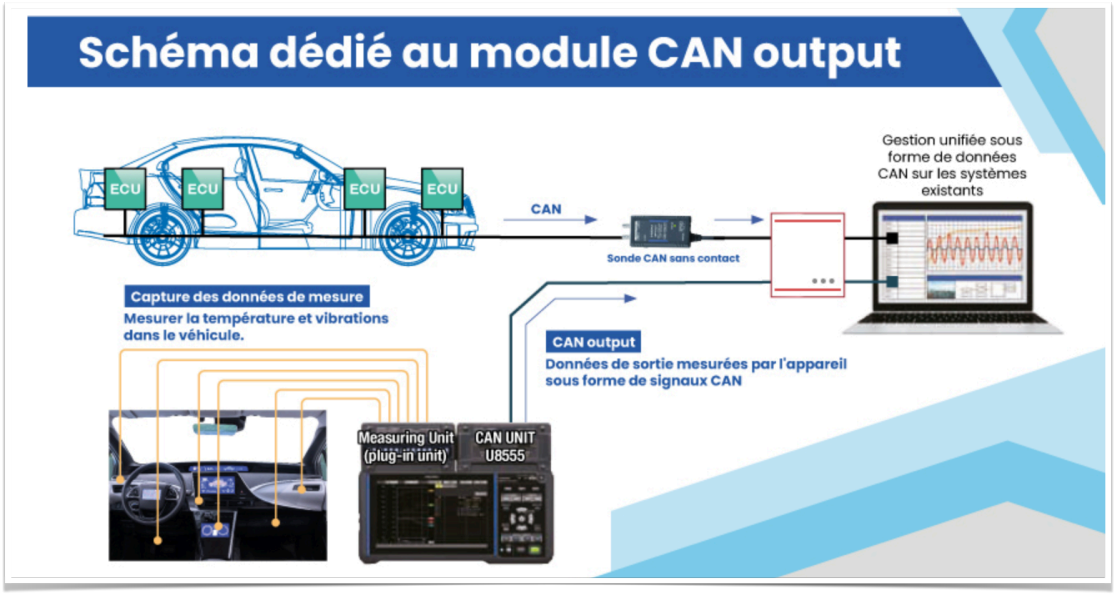
Reste donc à bien gérer la collecte de ces données en fonction du problème posé :
- mise en forme à partir d'un signal analogique [**numérisation**]
- quantité d'information à conserver [**échantillonnage**]

Résumé de la chaine de traitement d'un signal acoustique :



On peut souhaiter revenir à un signal analogique en bout de chaîne, en particulier sur les signaux nous sont destinés :
—>malgré tous les avantages du numérique l'être humain n'en demeure pas moins analogique …

<u>Exemple concret :</u>



<u>Le traitement par l'électronique embarquée d'une voiture passe par :</u>

- L'acquisition de données analogiques (capteur)
- La conversion CAN en données numériques
- Le traitement informatique (logiciel) des données
- ⇒ La réaction approprié du véhicule

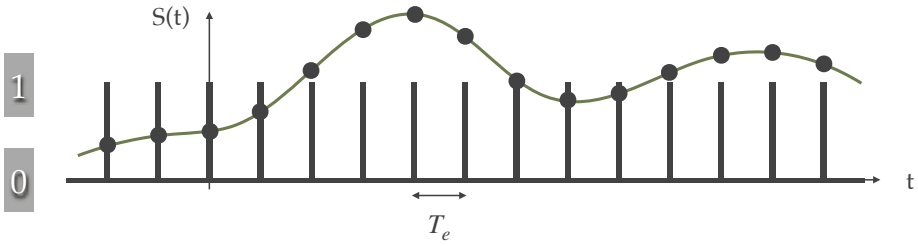
I - Echantillonnage et Quantification

1 - Echantillonnage : discrètisation temporelle

L'échantillonnage consiste à résumer le signal continu réel par des échantillons mesurés à intervalles réguliers T_e

Soit $f_e = \frac{1}{T_e}$ la fréquence d'échantillonnage.

On peut aussi interpréter l'échantillonnage comme le produit du signal avec un « peigne de Dirac » de période d'échantillonnage T_e :



On ajuste le nombre N_e en fonction de la résolution temporelle souhaiter, pour rester plus ou moins fidèle au signal initial. Naturellement, la taille en mémoire sera proportionnelle à N_e . Soit Δt la durée d'acquisition :

$$N_e = \frac{\Delta t}{T_e}$$

2 - Quantification : discrètisation des grandeurs

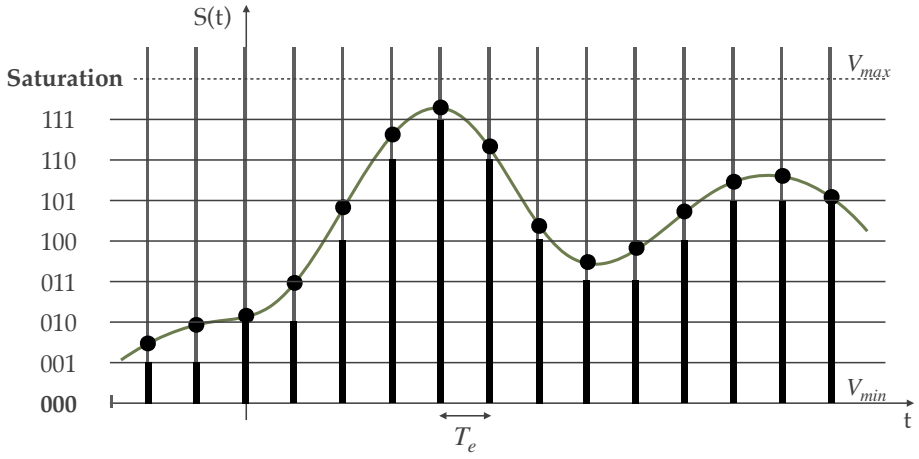
Les grandeurs physiques évoluent sur une plage continue. Le stockage numérique des données nécessite de résumer cette information quantitative, valeur de la grandeur à l'instant t, sous la forme d'une séquence binaire de taille donnée.
La taille en question : codage sur 8 bits sur 12 bits etc .. définit la précision mais également la place occupée en mémoire.

En général la carte d'acquisition propose de découper une plage de tension ΔV en 2^n pour un codage sur n bits.

Soit le pas de quantification : $q = \frac{\Delta V}{2^n}$

Soit $\Delta V = V_{max} - V_{min}$

Les valeurs restituées seront comprises entre :
V_{min} et $V_{min} + (2^n - 1) \times q = V_{max} - q$



De sorte que : - Tout signal inférieur à $V_{min} + q$ sera codé 000…00 et vaudra V_{min}
- Tout signal supérieur à $V_{max} - q$ sera codé 111…111 et vaudra $V_{max} - q$ ⇒ **Saturation**

Exercice : Calculer la taille en octet d'une mesure à l'oscilloscope sur 32 bits d'un échantillon de 10 ms, si on échantillonne le signal toutes les $1\mu s$.

Paramètres de l'acquisition :

- La durée d'acquisition Δt
- Le nombre d'acquisition N_e
- La période d'échantillonnage T_e

Ces trois paramètres ne sont pas indépendants

Relations :

On a : $T_e = \frac{\Delta t}{N_e}$ ou $\Delta t = N_e \times T_e$

ou $N_e = \frac{\Delta t}{T_e}$

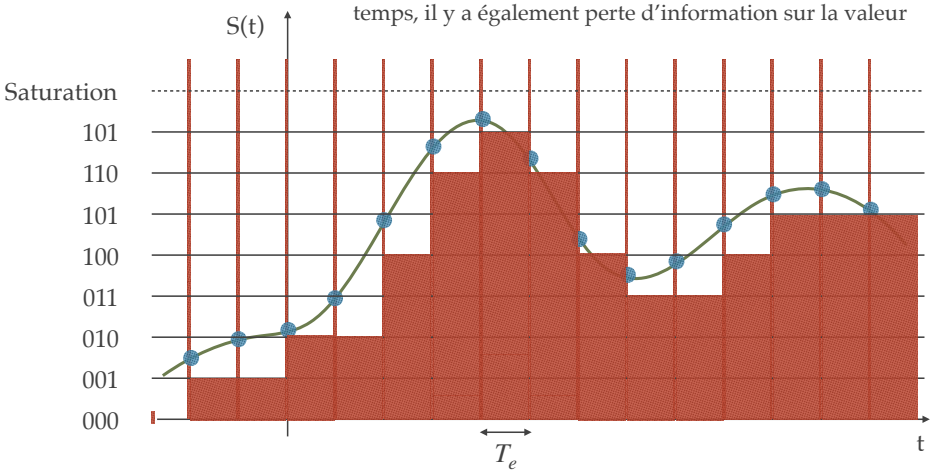
On définit la fréquence d'échantillonnage f_e par :

$$f_e \equiv \frac{1}{T_e}$$

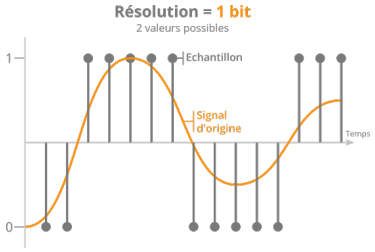
Ou encore : $f_e = \frac{N_e}{\Delta t}$

Nous avons à présent 3 paramètres liés entre eux

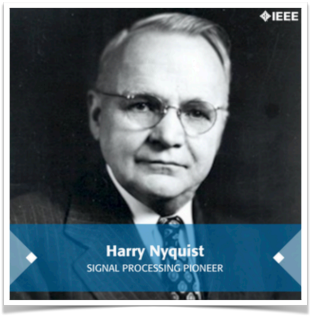
On peut voir que s'il y a eu perte d'information dans le temps, il y a également perte d'information sur la valeur



On peut voir que s'il y a eu perte d'information dans le temps, il y a également perte d'information sur la valeur



II - Condition de Nyquist - Shannon



1 - Théorème

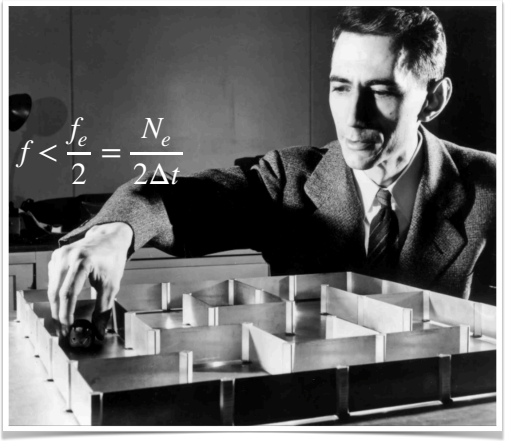
Théorème de Nyquist - Shannon : La restitution correcte d'un signal impose de l'échantillonner avec une fréquence au moins 2 fois supérieure à la fréquence maximale dans le spectre du signal.

Interprétations :

- Cela permet de suivre les variations de toutes les composantes spectrales.
- Il faut a minima résoudre le min et le max de la composante la plus rapide : $f_e = 2\,f_{max}$
- Il faut éviter le phénomène de repli spectral (sous-échantillonnage).

Conséquence :

Pour un signal périodique quelconque, il y a en générale une infinité d'harmoniques.
La condition de Nyquist- Shannon est impossible à respecter. Elle se traduira plutôt par une incapacité à retranscrire proprement toutes les fréquences supérieures à $f_e/2$.
Exemple : phénomène de repli de spectre.



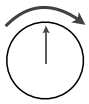
Cette nouvelle contrainte se traduit sur les fréquences mesurables pas l'échantillonneur :

$f < \frac{f_e}{2}$ $\quad$ $f < \frac{1}{2\,T_e}$ $\quad$ $f < \frac{N_e}{2\Delta t}$ $\quad$ **Critère de Shannon**

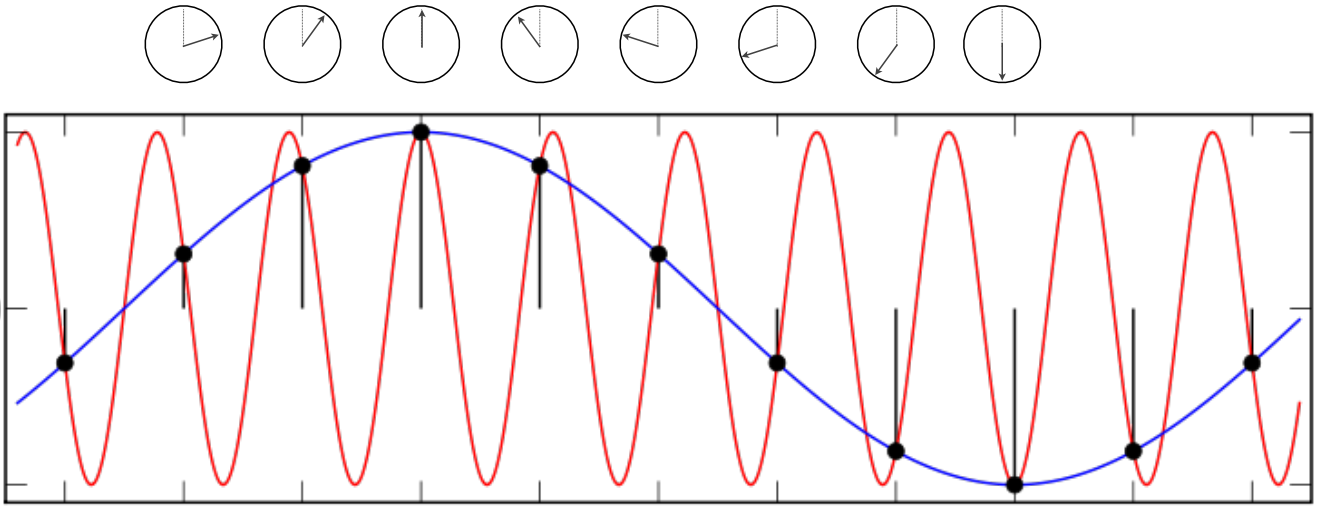
Si cette condition n'est pas respectée, les fréquences supérieures à $f_e/2$ ne seront pas mesurables, mais pire encore elles vont spolier les mesures aux basses fréquences.

2 - Effet stroboscopique de sous-échantillonnage :

Cet effet a été vu en TP au stroboscope ou sur une vidéo où des roues de voiture semblent tourner à l'envers.
On en voit la raison sur le schéma ci-dessous :
Entre deux images la roues fait presque son tour mais pas tout à fait, ainsi elle semble tourner à l'envers.
Quelle est sa fréquence apparente ?



Sens réel
de rotation



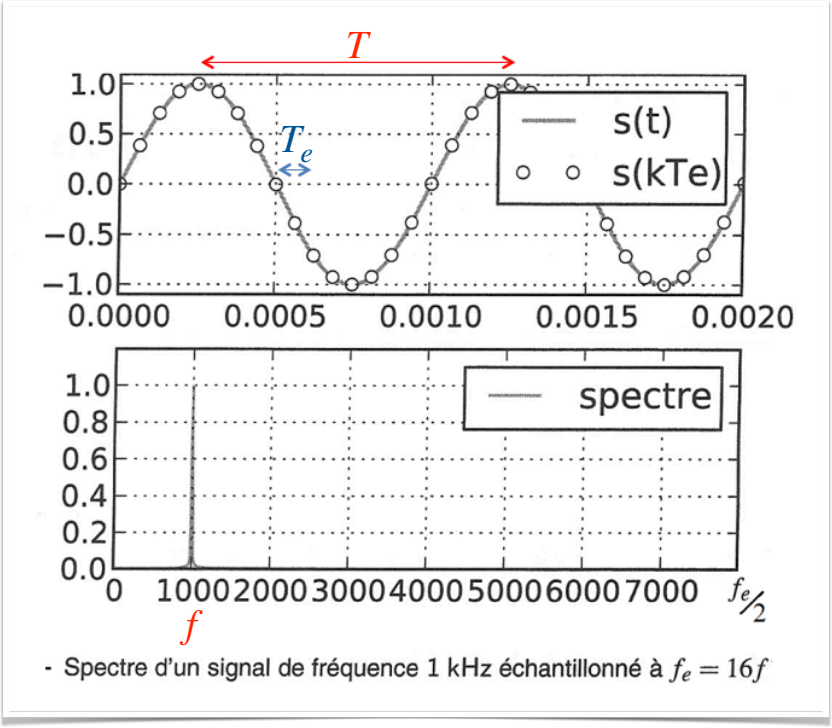
Soit $\omega = 2\pi f$ la pulsation réelle du disque et soit $\omega_e = 2\pi f_e$ la pulsation d'échantillonnage. **Calculer la pulsation apparente** $\omega_a = 2\pi f_a$

Méthode :

- On établit l'angle apparent dont à tourner le disque **[modulo** 2π**]** —> That's the point boys !
- On en tire la/les pulsations possibles. Noter que certaines sont négatives. Rotation inversée.
- En injectant ces pulsations dans un signal sinusoïdal, en déduire **toutes** les fréquences engendrées par l'échantillonnage. **Quel que soit leur signe**.
  Rq : $cos(x) = cos(-x)$ et on ne tient pas compte ici du déphasage.

Exemple : Signal harmonique bien échantillonné

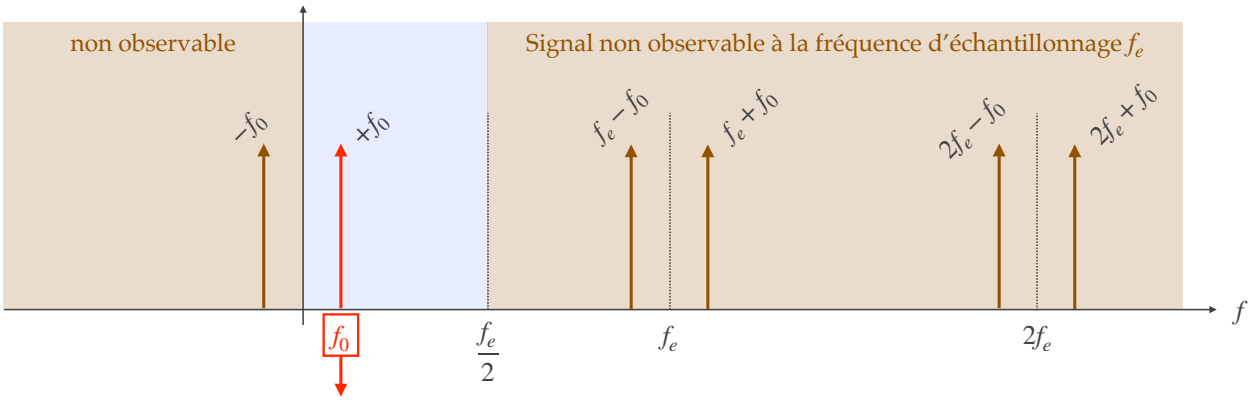
On observe bien un unique pic à la fréquence $f = 1kHz$



- Spectre d'un signal de fréquence 1 kHz échantillonné à $f_e = 16f$

3 - Conséquence sur le spectre : On génère toutes les fréquences : $f = nf_e \pm f_0$ avec $n \in \mathbb{Z}$

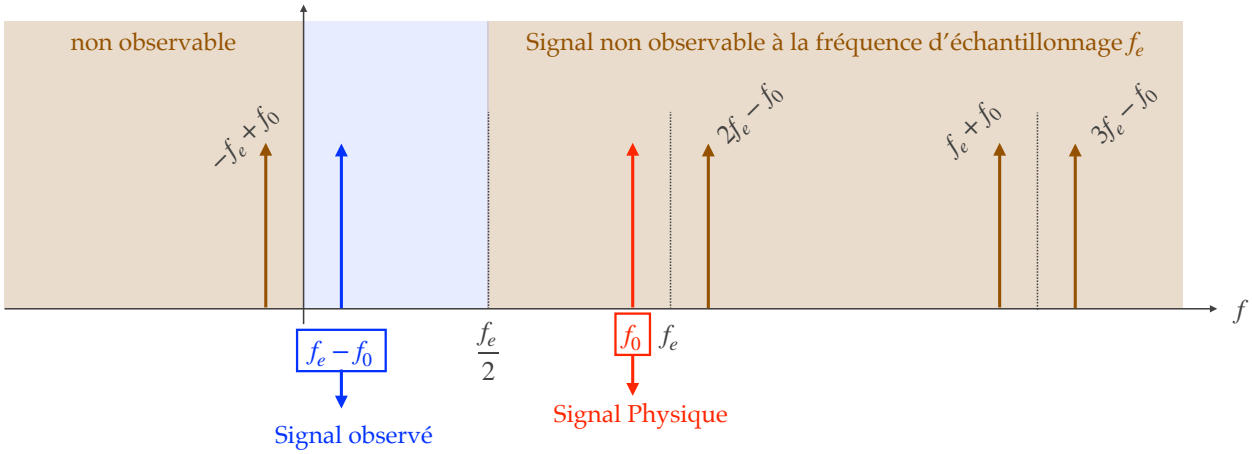
Exemple de signal bien échantillonné :



- Le signal observé est bien le signal physique
- Les composantes parasites n'entrent pas dans la plage de représentation de l'échantillon.

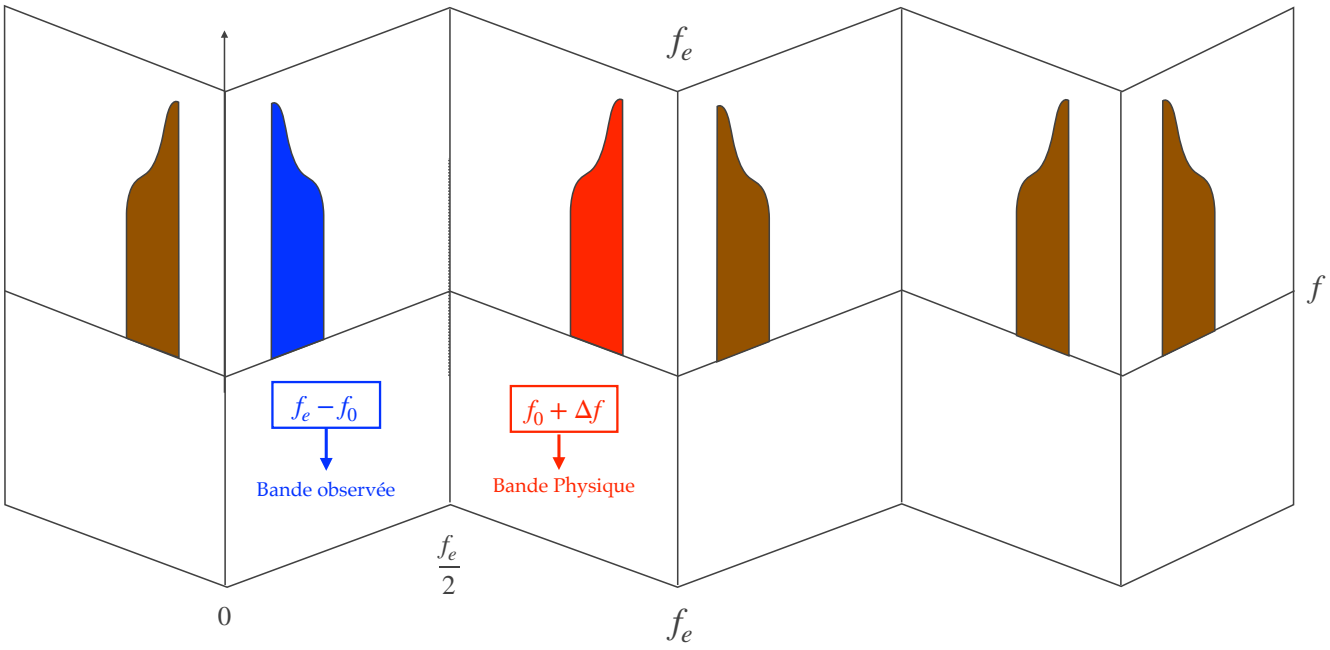
Conséquence sur le spectre : On génère toutes les fréquences : $f = nf_e \pm f_0$

Exemple de signal sous - échantillonné :



C'est le cas de la roue qui semble tourner à l'envers

Effet de bande On a un **effet de repli en accordéon**

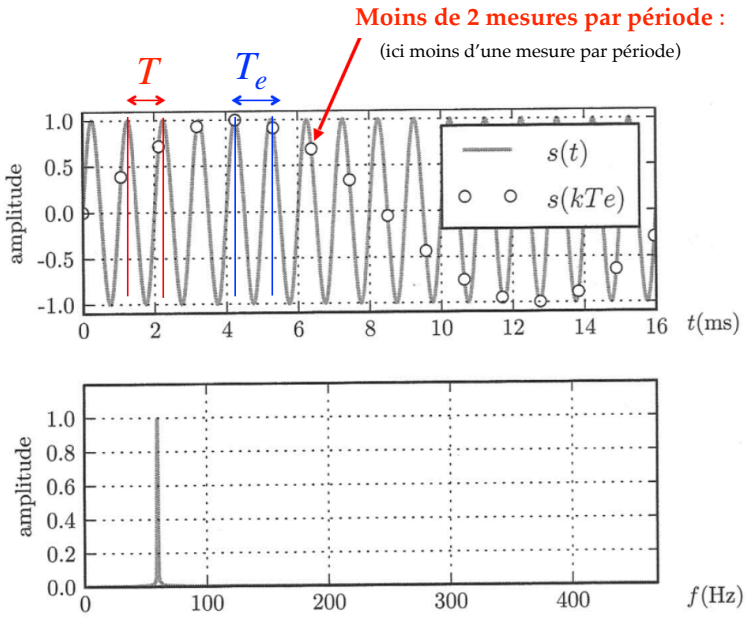


Exemple : Signal harmonique sous-échantillonné

Exercice :

Rien n'indique visuellement le problème car on a bien un pic mais la fréquence n'est pas la bonne !

- A vous de justifier la fréquence apparente f_a
- Qu'en serait-il si $T_e = \frac{17}{18}T$?
- Trouver T_e pour que l'échantillonnage soit convenable

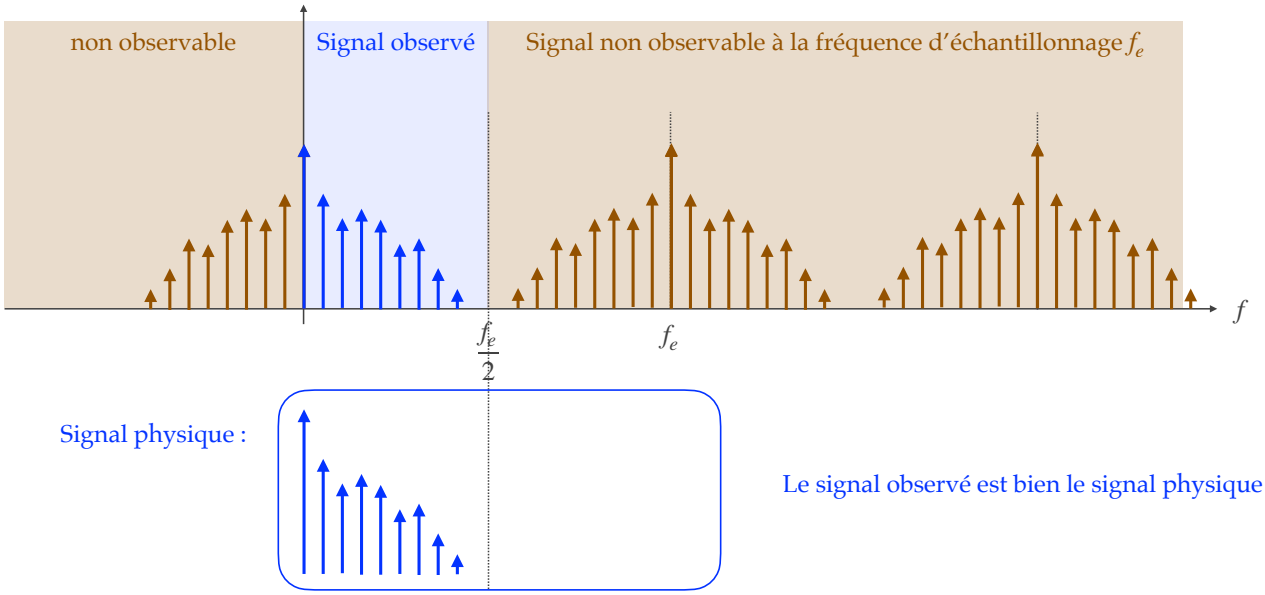


$f_a = \frac{f}{17} = 58.8\ Hz$

Signal de fréquence 1 kHz échantillonné à $T_e = \frac{17}{16}T$.

Conséquence sur le spectre : On génère toutes les fréquences : $f = nf_e \pm f_0$

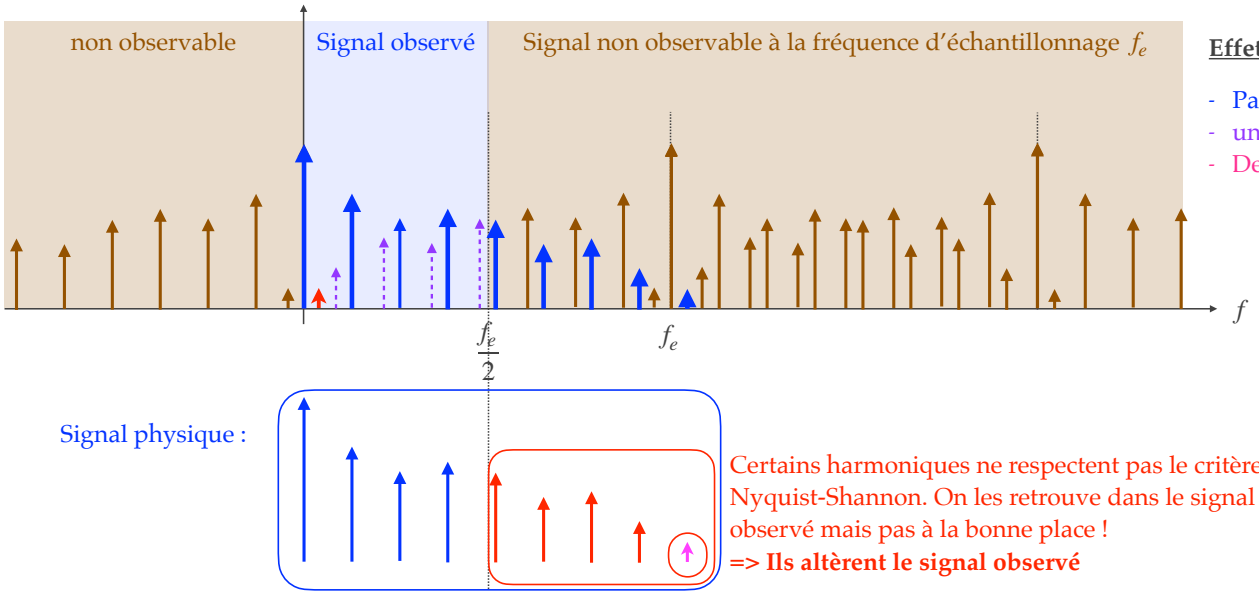
Exemple de signal **bien - échantillonné** :



Le signal observé est bien le signal physique !

Conséquence sur le spectre : On génère toutes les fréquences : $f = nf_e \pm f_0$

Exemple de signal sous - échantillonné :



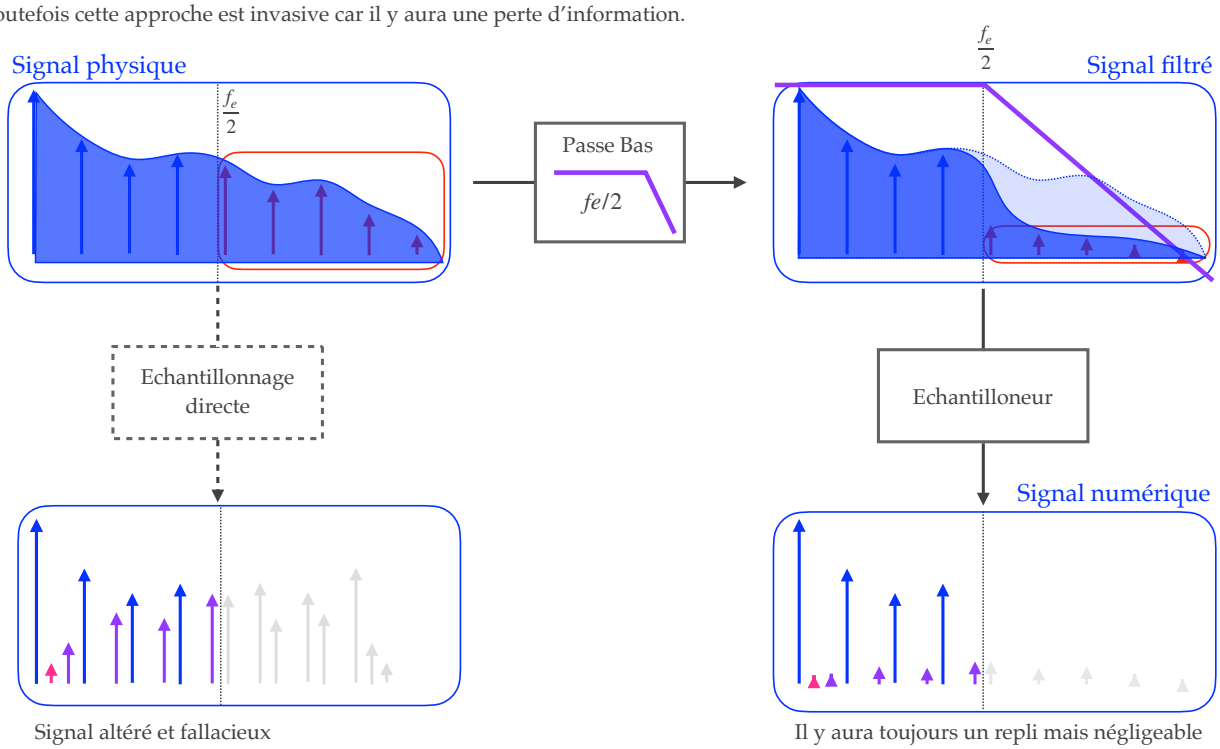
Effet en accordéon :

- Pas de repli
- un repli
- Deux replis

Certains harmoniques ne respectent pas le critère de Nyquist-Shannon. On les retrouve dans le signal observé mais pas à la bonne place !
=> Ils altèrent le signal observé

4 - Filtrage analogique :

Une « solution » pour éviter le repli de spectre de filtrer le signal physique avant l'échantillonnage avec un Passe Bas analogique qui coupe à *fe*/2. Toutefois cette approche est invasive car il y aura une perte d'information.



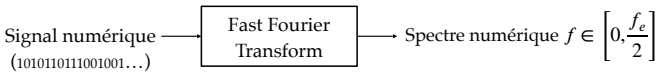
Signal altéré et fallacieux

Il y aura toujours un repli mais négligeable

III - Analyse spectrale numérique

Transformée de Fourier discrète d'un signal numérique :

Une fois réalisé l'échantillonnage, un algorithme dit FFT pour « Fast Fourier Transform » permet de calculer spectre discret associé à l'échantillonnage.

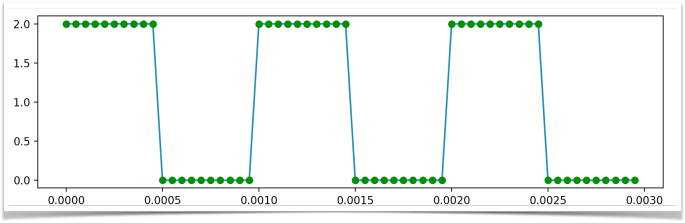


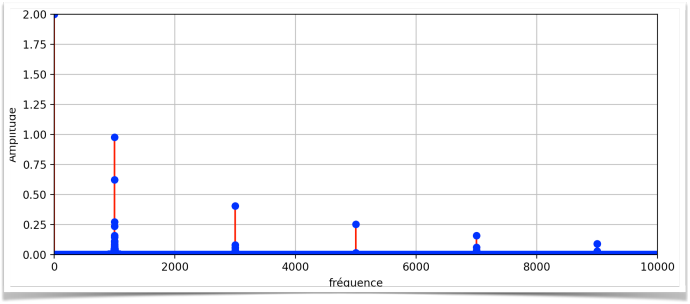
Soit N_e le nombre d'échantillons sur une durée Δt, on a donc :

$$T_e = \frac{\Delta t}{N_e} \quad \text{ou encore} \quad f_e = \frac{N_e}{\Delta t}$$

L'algorithme FFT renvoie N_e valeurs spectrales équi-réparties sur la plage $f \in [0, f_e]$. On en déduit la résolution spectrale :

$$\Delta f = \frac{f_e}{N_e} = \frac{N_e}{N_e \times \Delta t} = \frac{1}{\Delta t} \quad \text{soit} \quad \Delta f = \frac{1}{\Delta t}$$

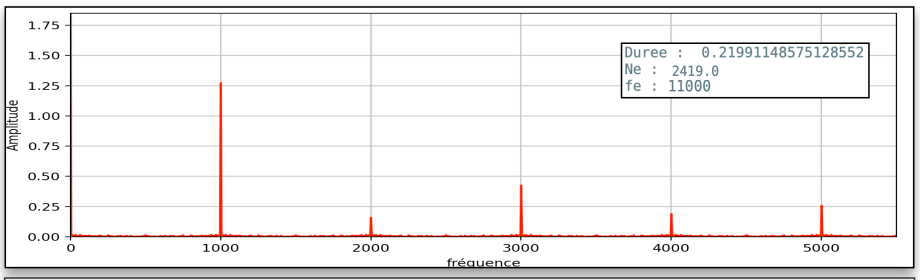


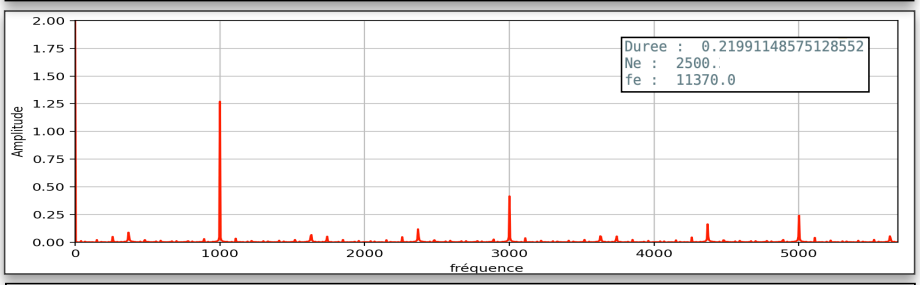


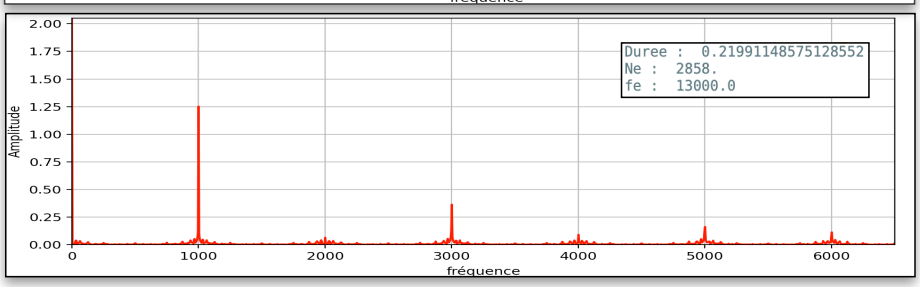
1 - On choisit d'abord la plage spectrale $f \in [0, f_e/2]$ en fonction du spectre. Cela revient à choisir T_e

2 - On ajuste la durée d'acquisition Δt pour contrôler la résolution spectrale.

On procèderai de la même façon avec l'oscilloscope.







Effet de la fréquence d'échantillonnage :

Lorsque le signal est T-périodique, il y a théoriquement une infinité de composantes spectrales. Toutefois celles-ci décroissent rapidement.

On peut donc choisir f_e telle que toutes les composantes spectrales au dessus de $f_e/2$ deviennent négligeables. Ainsi le spectre sera valide malgré l'effet de repli.

Exemple :

On échantillonne ici autour de 11 $f \sim 11\ kHz$ et on coupe donc à ~5500 Hz.

Dans tous les cas il y a des replis, mais **mieux vaut éviter que les replis ne tombent :**

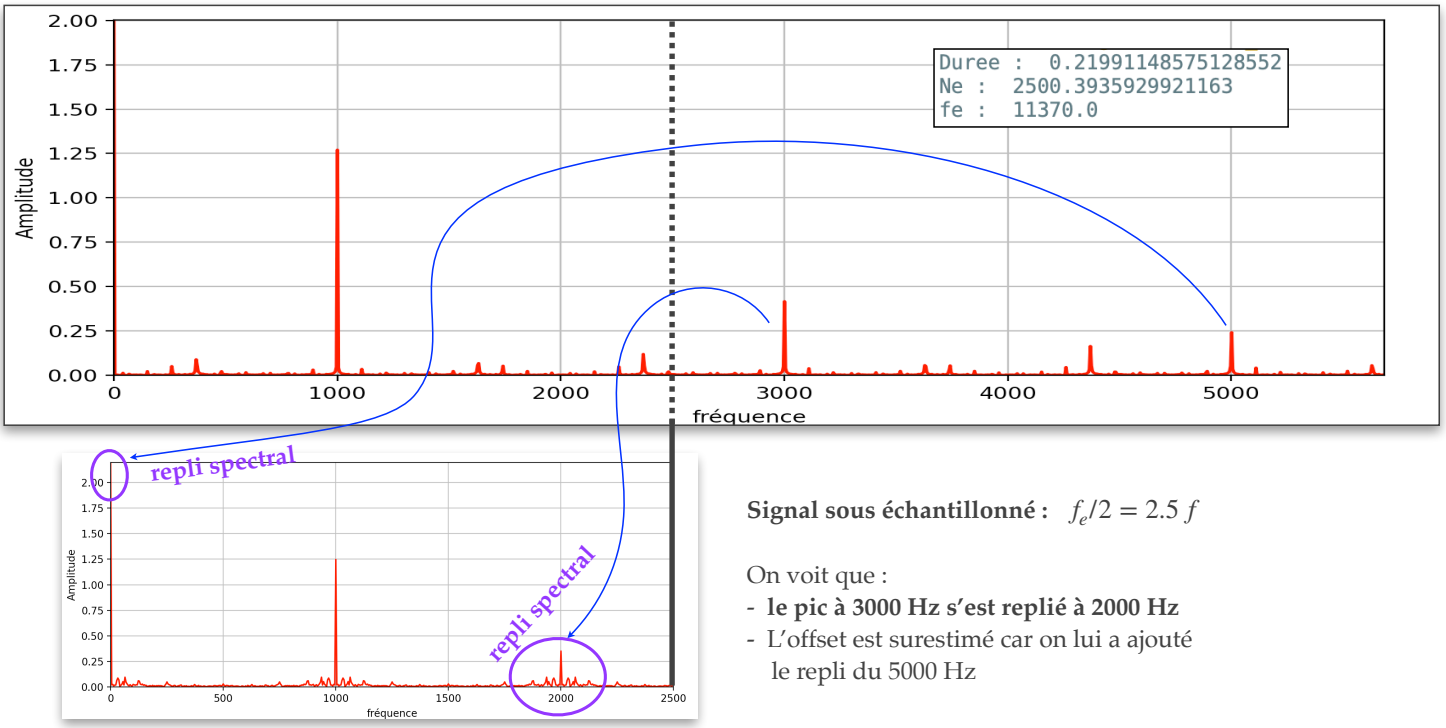
- **sur des composantes existantes** :
  Erreur de mesure de l'amplitude

- **sur des multiples du fondamental** :
  Ce qui fait apparaitre des pics fictifs.

Si la capacité de calcul le permet on prendra :

$$f_e/2 \gg f$$

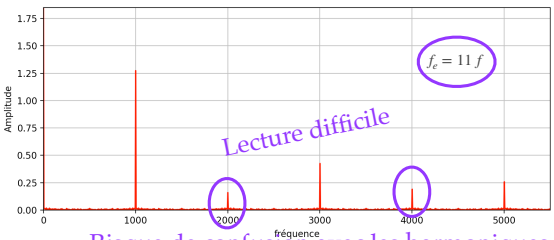
Signal assez bien échantillonné : $f_e/2 \sim 5f$ on voit bien les 3 premiers harmoniques. [Rq : f_e n'est pas un multiple de f]



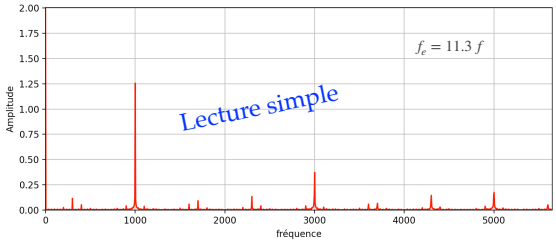
Signal sous échantillonné : $f_e/2 = 2.5\ f$

On voit que :
- **le pic à 3000 Hz s'est replié à 2000 Hz**
- L'offset est surestimé car on lui a ajouté le repli du 5000 Hz

Rq 1 : Ajustement plus fin de la fréquence d'échantillonnage :

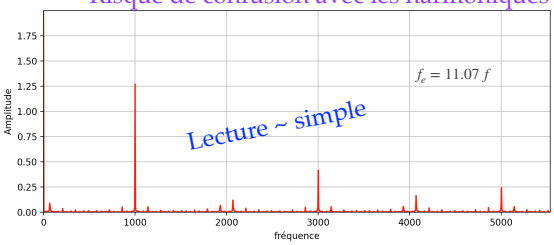


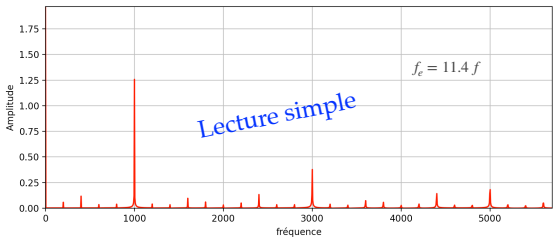
Risque de confusion avec les harmoniques

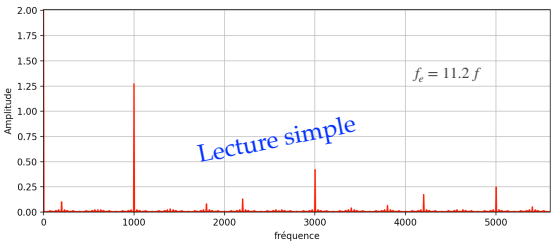


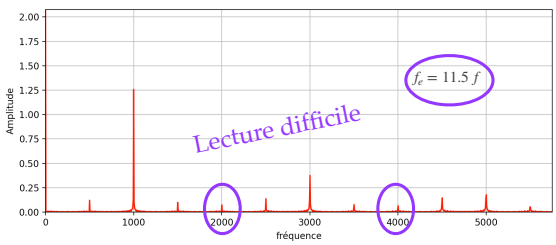
Les petits « picots » ne tombent pas sur des harmoniques.

⇒On identifie les replis sans risque de confusion.



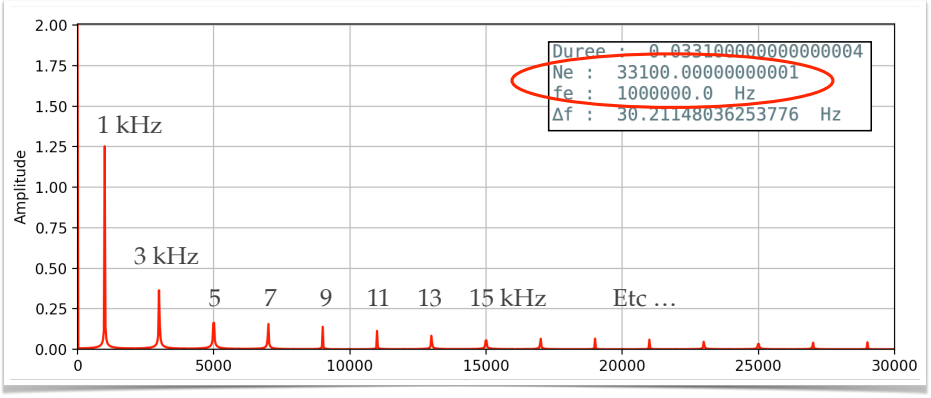


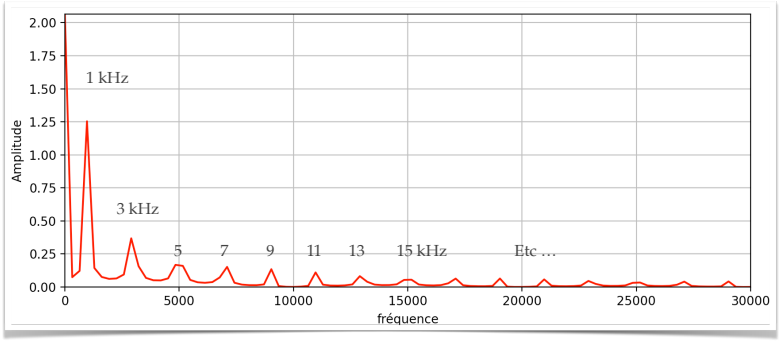




Eviter les multiples
ou
Les fractions simples

Rq 2 : Pas de difficulté avec $f_e = 1000\ f$!!! Soit $f_e/2 = 500.000\ Hz$ $\Rightarrow$ On ne risque pas de voir un repli !
Toutefois une bonne partie du spectre calculé est sacrifiée (même pas affichée ici)

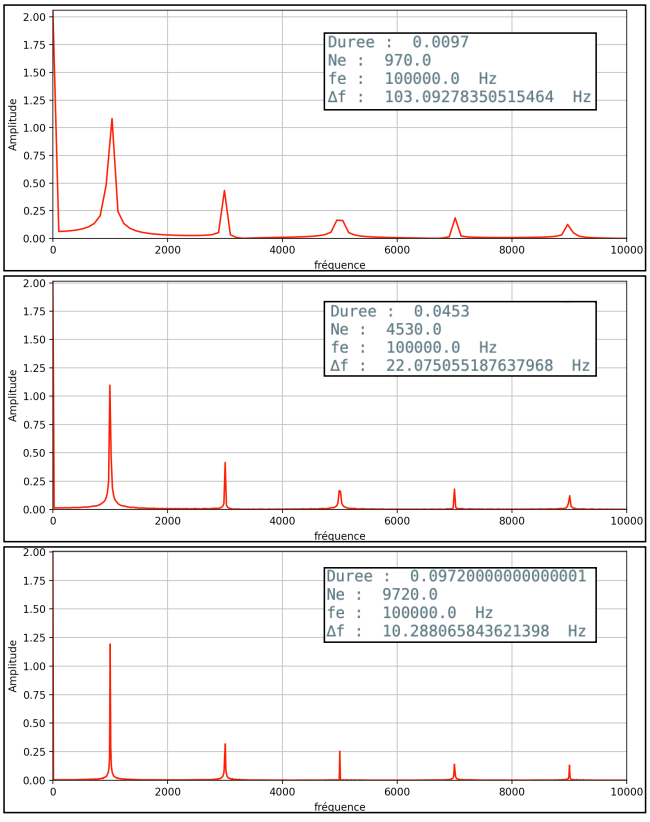


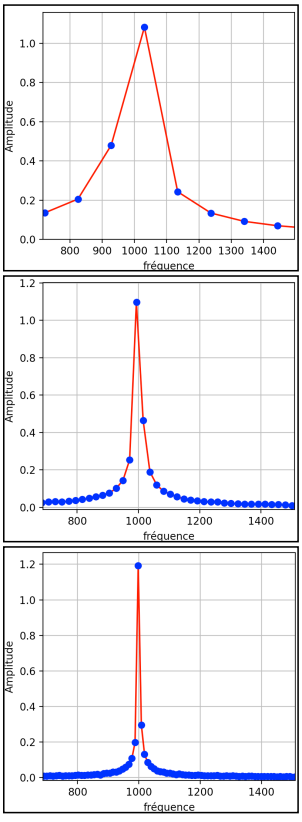


Le spectre est débarrassé des replis !

En général, la résolution ne sera pas bonne si on a juste augmenté f_e sans changer d'autre paramètre. [voir ci-contre]

Il nous faut désormais améliorer la résolution.





Effet de la durée :

La résolution en fréquence est égale à l'inverse de la durée : $\Delta f = \dfrac{1}{\Delta t}$

1 - On fixe la fréquence d'échantillonnage assez haut pour négliger l'effet de repli.

2 - On augmente la durée pour améliorer la résolution spectrale.

C'est ainsi que l'on procède à l'oscilloscope.

Effet du nombre de points :

$$f_e = \frac{N_e}{\Delta t} \qquad \text{ou} \qquad \Delta t = N_e \times T_e$$

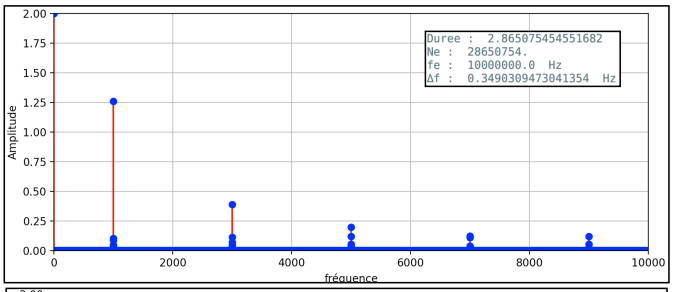
Les trois paramètres étant liés, jouer sur le nombre de points N_e revient à jouer :
- sur f_e si on a déjà fixé Δt
- sur Δt si on a déjà fixé T_e c-à-d f_e

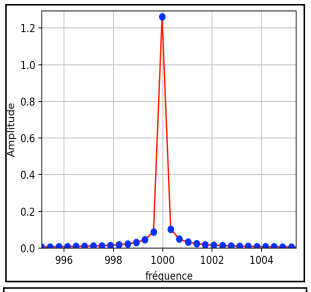
On note enfin que **la valeur théorique des pics est très approximative et fluctue au moindre changement des paramètres**.
Cela s'explique du fait que :

- Le créneau présente une discontinuité qui n'est pas bien restituée par l'échantillonnage .
- la courbe est très piquée, donc la valeur maximale dépend beaucoup de la position des points autour de la fréquence théorique.

En pratique à l'oscilloscope les amplitudes fluctuent aussi au cours du temps car le spectre est recalculé en permanence et l'échantillonnage exact varie en fonction de la synchronisation de la base de temps.

En théorie les coefficients de ce créneau seraient : (2 offset) puis : 1 - 0.333 - 0.2 - 0.143 - 0.111 etc… soit les inverses des impairs.
En revanche **une fois réglé les effets de repli, la position relative en fréquence des pics est excellente : C'est là ce que l'on recherche !**

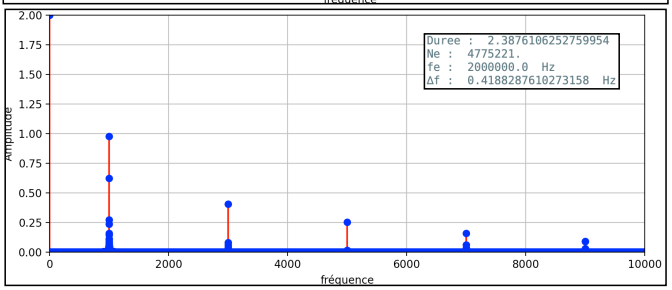


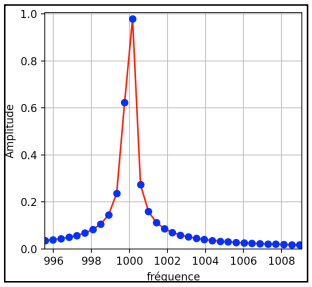


Un peu plus précis et pourtant : **incorrect !**

—> On constate que le pic est mal résolut

Rq : Le 3ème harmonique est réaliste
C'est assez aléatoire.





Un peu moins précis et pourtant ~ **correct !**

—> On constate que le pic est bien mieux résolu

Rq : ici le 2ème harmonique est incorrect

Le programme de calcul de la FFTD

```
import math
import numpy as np
from matplotlib.pyplot import *
from numpy.fft import fft


f=1e3
T=1/f;    a=2.;     phi=0.;   offset=0.


def signal(t):
    if t%T<T/2: return a
    else:       return 0  #creneau
```

```
## Echantillonage
Duree = 33.1*T

fe = 1000*f
Te = 1/fe
Ne = Duree/Te

t = np.arange(start=0.0,stop=Duree,step=Te)
echantillons = [signal(x) for x in t]

"""
plot(t,echantillons)       #Tracé du signal
plot(t,echantillons,"go")
show()
"""
print("Duree : ",Duree)
print("Ne : ",Ne)
print("fe : ",fe," Hz")
print("Δf : ",1./Duree," Hz" )
```

```
##Calcul du spectre
tfd = fft(echantillons)
N=len(echantillons)
spectre = np.absolute(tfd)*2/N

#ATTENTION : L'échelle des fréquences est construite sachant que l'espacement
#            fréquentiel de deux points de la TFD est l'inverse de la durée T :
#Soit :

freq=np.arange(N)*1.0/Duree
```

```
##Affichage :

figure(figsize=(10,4))
plot(freq,spectre,'r')
#plot(freq,spectre,'bo')              #ajout des points
xlabel('fréquence')
ylabel('Amplitude')
#axis([-0.1,fe/2,0,spectre.max()])    #fenetre de Shannon
axis([-0.1,30*f,0,spectre.max()])     #10* f_signal  <<  fe /2  => pas de repli
grid()
show()
```

IV - Filtrage numérique

Notre signal étant échantillonné, comment réaliser sur ce signal les opérations de filtrage ordinaires ? (cf 1ère année)
Il existe deux approches au filtrage numérique : l'une fréquentielle l'autre temporelle.

1 - Filtrage numérique fréquentiel :

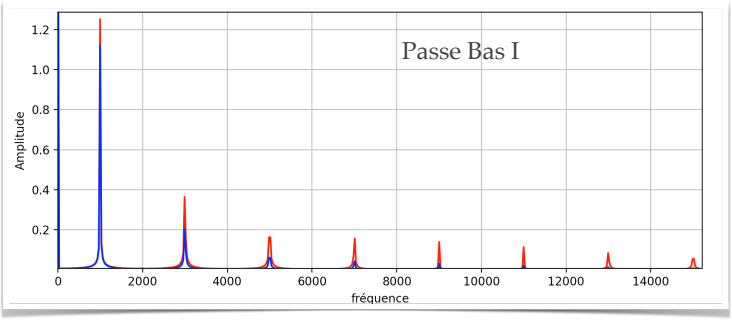
Il suffit de calculer le gain en fonction de la fréquence et de multiplier à chaque fréquence, le spectre discret par le gain de sa fréquence

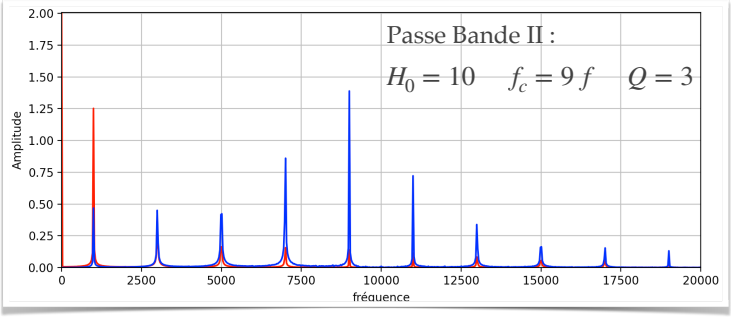
Exemple d'un passe bas du 1er ordre :

Exemple d'un passe bande du 2nd ordre :

Réalisation en Python :

Il suffit de calculer le gain en fonction de la fréquence et de multiplier à chaque fréquence, le spectre discret par le gain de sa fréquence






```
## FILTRAGE NUMERIQUE FREQUENTIEL

#on repart du spectre obtenu par fftd :  freq et spectre

def PBasI(f,spc,fc):
    G     = [1/(1+(x/fc)**2)**0.5 for x in f]
    spc_f = [G[k]*spc[k] for k in range(len(G))]
    return spc_f

fc=2e3#Hz
spec_filtre=PBasI(freq, spectre, fc)
```

```
figure(figsize=(10,4))
plot(freq,spectre,'r')
plot(freq,spec_filtre,'b')
xlabel('fréquence')
ylabel('Amplitude')
axis([-0.1,20*f,0,spectre.max()])
#10* f_signal  <<  fe /2  => pas de repli
grid()
```

```
def PBandeII(f,spc,H0=1, fc=1e3, Q=1):
    G     = [H0*x/(Q*fc)/(  (1-(x/fc)**2)**2 + (x/(Q*fc))**2)**0.5 for x in f]
    spc_f = [G[k]*spc[k] for k in range(len(G))]
    return spc_f

fc=9e3#Hz
H0 =10
Q  =3
spec_filtre=PBandeII(freq, spectre, H0, fc, Q)
```

2 - Filtrage numérique temporel :

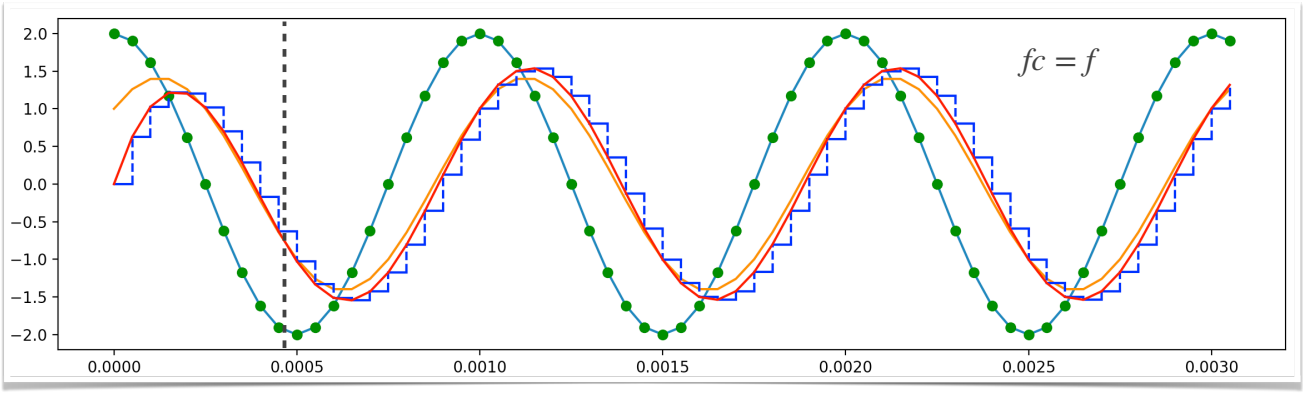
Soit $e(t)$ un signal en entrée d'un filtre et $s(t)$ le signal filtré. On suppose le régime transitoire terminé ; la relation entrée - sortie du filtre est régie par la fonction de transfert du filtre ou son équation différentielle équivalente.

Le filtrage temporel consiste à intégrer numériquement l'équation différentielle pour trouver l'expression de la sortie.
On procède par exemple par la méthode d'Euler.

Exemple d'un passe bas du 1er ordre :

Réalisation en Python :
On réalise une boucle d'intégration d'Euler

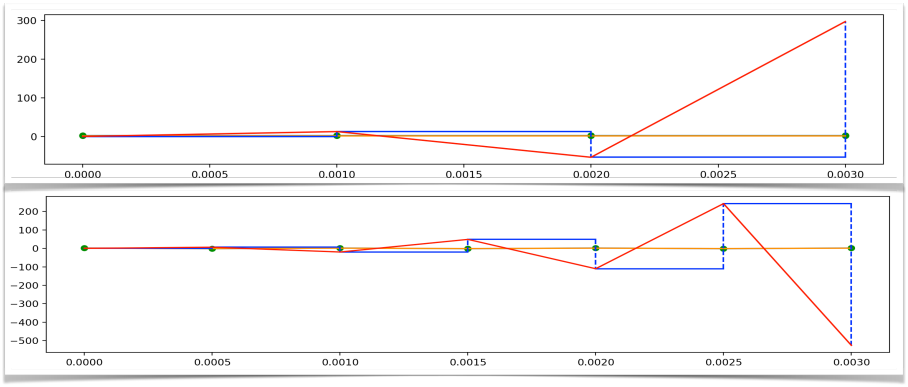
Régime « numériquement » Etabli



Régime Transitoire
(numériquement)

On observe que malgré une condition initiale quelconque, la solution filtrée par intégration converge rapidement vers le solution théorique [**intégration numériquement stable**]

On s'est placé arbitrairement à la coupure du filtre passe bas.



$f_e = f$

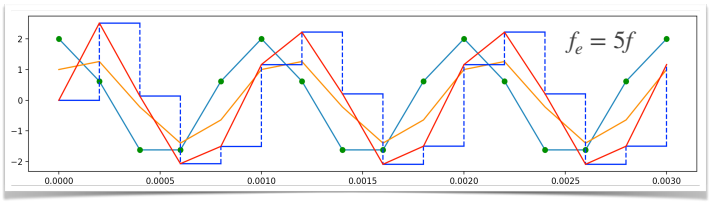
$f_e = 2f$

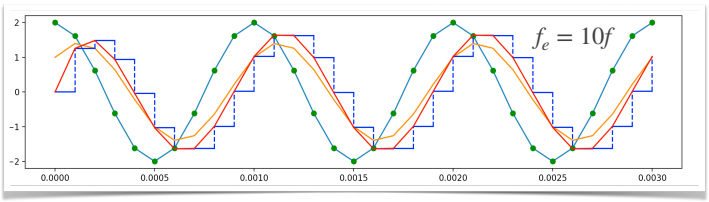
Régimes divergents :

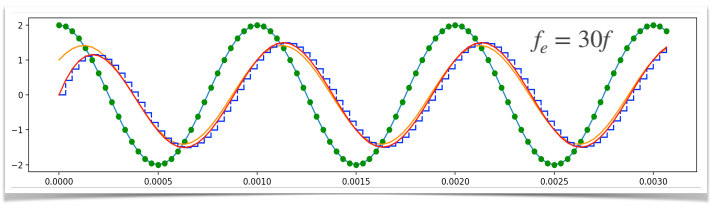
On peut confondre le signal avec un signal de plus basse fréquence.

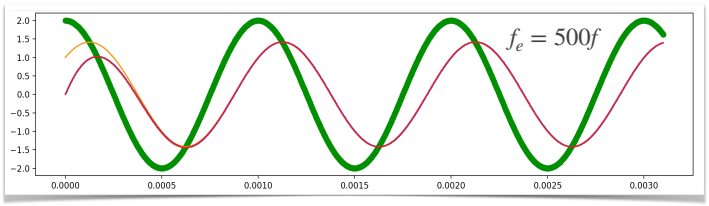
De plus, l'intégration d'Euler propage des erreurs qui croissent exponentiellement.

On veillera donc à se placer bien au delà du critère de Shannon.



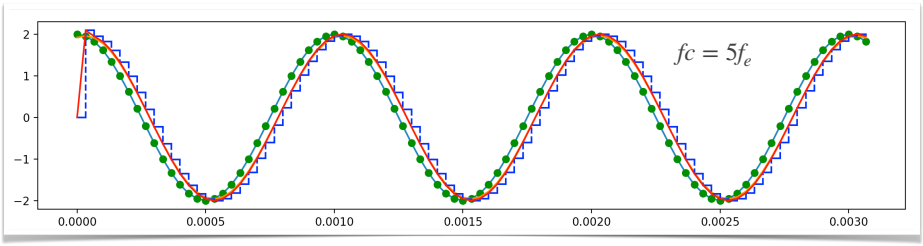


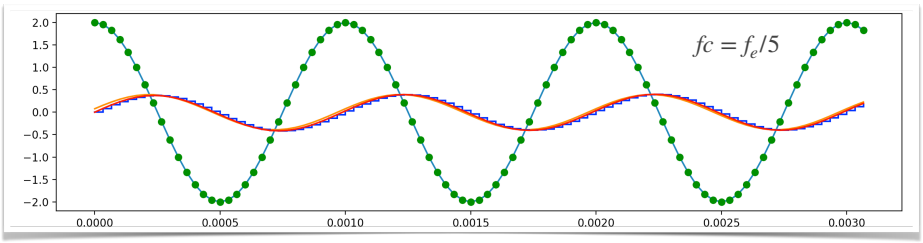


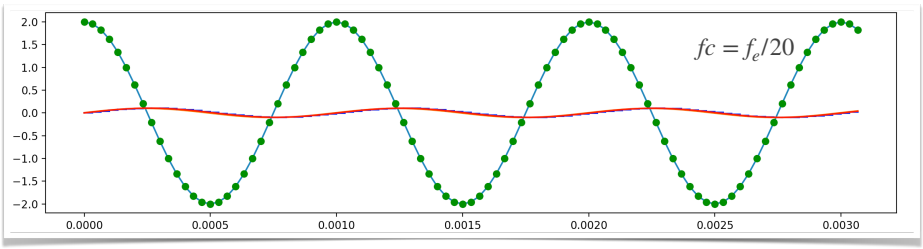


Evolution en fréquence :

On retrouve le résultat observé en TP à l'oscilloscope.





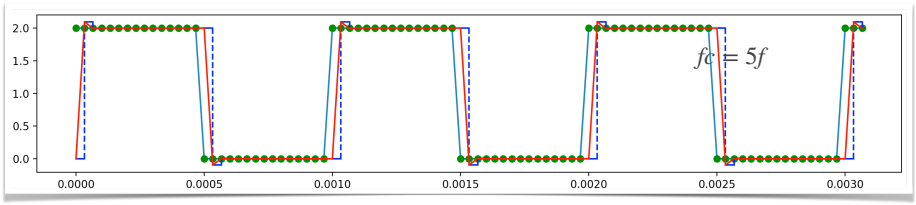


En particulier :

- Le gain tend vers 0
- Le déphasage vers une quadrature retard

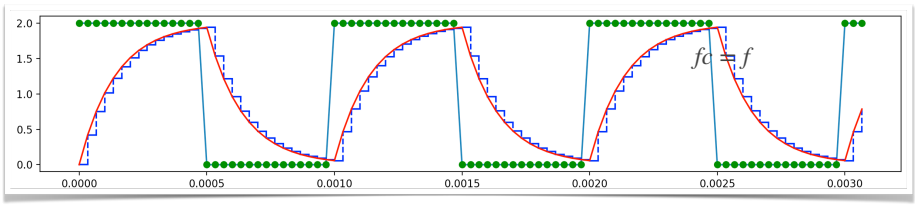
Filtrage d'un créneau :

Ayant validé notre filtre sur la solution théorique d'un signal harmonique, on peut l'essayer sur un signal T-périodique.
A nouveau la solution est conforme aux observations expérimentales.

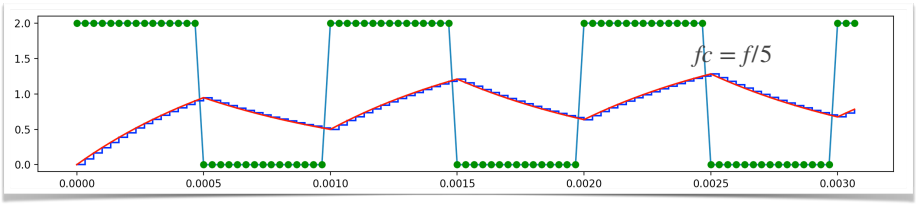


Problème dans les 3 cas :
Il est difficile de restitué la discontinuité …

« paradoxe de Gibbs »



On retrouve le régime :
Charge-décharge



On retrouve le comportement :
pseudo-intégrateur

Filtrage numérique en python

```python
#filtre PBasI de gain H0, coupure fc
H0=1; fc=f/5
G   =  H0 / (1+(f/fc)**2)**0.5  #paramètre du filtre en fonction de f et fc
psi = -atan(f/fc)

def signalFiltre_TH(t):

    return G*a*np.cos(2*pi/T*t + phi+ psi) #amp*Gain   & phase + dephasage
```

```python
## Echantillonage
Duree = 3.1*T

fe = 30*f
Te = 1/fe
Ne = Duree/Te

t = np.arange(start=0.0,stop=Duree,step=Te)
echantillons    = [signal(x)          for x in t]  #signal echantilloné
echantillons_th = [signalFiltre_TH(x) for x in t]  #signal filtré théorique --> tracé
```

```python
## Affichage
plot(t,echantillons)       #Tracé du signal
plot(t,echantillons,"go")
plot(t,echantillons_th)    #Tracé du signal filtré théorique
```

```python
## FILTRAGE NUMERIQUE TEMPOREL


e=echantillons
s=[0]               #Cond. Ini. arbitraire

for i in range(len(t)-1):
    plot([t[i],t[i]+Te],[s[i],s[i]],"b") #palier horizontal

    ds = 2*pi*fc*(e[i]-s[i])*Te          #Méthode Euler
    s+=[s[i] + ds ]

    plot([t[i+1],t[i+1]],[s[i],s[i+1]],"b--") #dash-line verticale


plot(t,s,"r") #signal filtré interpolé
show()
```