Échantillonnage et reconstruction d'un signal périodique

1. Introduction

L'échantillonnage d'un signal continu est l'opération qui consiste à prélever des échantillons du signal pour obtenir un signal discret, c'est-à-dire une suite de nombres représentant le signal, dans le but de mémoriser, transmettre, ou traiter le signal.

L'échantillonnage intervient dans l'opération de conversion analogique-numérique, par exemple dans un dispositif de numérisation du son ou de l'image. Un autre exemple d'échantillonnage est celui que l'on fait pour obtenir la représentation graphique d'une fonction à une ou deux variables. D'une manière générale, l'échantillonnage intervient dans toute opération de conversion continu/discret.

Ce document présente le théorème de l'échantillonnage de Shannon, qui permet de savoir à quelle fréquence minimale il faut échantillonner un signal pour ne pas perdre l'information qu'il contient. On verra aussi comment se fait la reconstruction d'un signal continu à partir des échantillons, opération qui intervient dans la conversion numérique-analogique.

Pour expliquer l'échantillonnage et la reconstruction, il faut utiliser l'analyse spectrale et la transformée de Fourier discrète, abordées dans le document Introduction à l'analyse spectrale.

On s'intéressera à un signal temporel représenté par une fonction $u(t)$, où t est le temps, mais les résultats se transposent sans difficulté aux cas de fonctions d'autres variables, par exemple de variables d'espace. En particulier, les résultats seront utilisables pour l'échantillonnage d'une image, c'est-à-dire une fonction $I(x, y)$ de deux variables d'espace.

Le signal $u(t)$ considéré est supposé périodique. En pratique, l'échantillonnage de ce signal se fait nécessairement sur une durée finie. En toute rigueur, le signal que l'on échantillonne n'est pas donc pas périodique (on peut le qualifier de quasi périodique).

2. Échantillonnage

2.a. Théorème de Shannon

Soit $u(t)$ une fonction représentant un signal continu. On considère un échantillonnage périodique défini par :

$$t_k = kT_e \tag{1}$$
$$u_k = u(t_k) \tag{2}$$

où k est un nombre entier. T_e est la période d'échantillonnage. $f_e = 1/T_e$ est la fréquence d'échantillonnage.

Le théorème de Shannon ([1]) concerne les signaux dont le spectre possède une fréquence maximale f_{max}, que l'on appelle des signaux à bande limitée. Par exemple, si $u(t)$ est un polynôme trigonométrique, la fréquence maximale est celle de la plus grande harmonique.

Théorème de Shannon : pour que le signal puisse être entièrement reconstruit à partir des échantillons, il faut et il suffit que :

$$f_e > 2f_{max} \tag{3}$$

La fréquence d'échantillonnage doit être strictement supérieure à deux fois la plus grande fréquence présente dans le spectre du signal continu (condition de Nyquist-Shannon). Si cette condition est vérifiée alors :

$$u(t) = \sum_{k=-\infty}^{+\infty} u_k \text{sinc}\left(\frac{t-kT_e}{T_e}\right) \tag{4}$$

où la fonction sinus cardinale est définie par :

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x} \tag{5}$$

Cette relation montre que le signal peut être reconstruit à partir des échantillons, ce qui signifie que toute l'information présente dans le signal original est conservée dans les échantillons. Nous verrons plus loin comment l'opération de reconstruction est effectuée en pratique.

La moitié de la fréquence d'échantillonnage est appelée la fréquence de Nyquist f_n et la condition de Nyquist-Shannon s'écrit donc $f_{max} < f_n$.

Lorsque la condition n'est pas vérifiée, on dit qu'il y a sous-échantillonnage. On parle de sur-échantillonnage lorsque la fréquence de Nyquist est beaucoup plus grande que f_{max}.

2.b. Fonction sinusoïdale

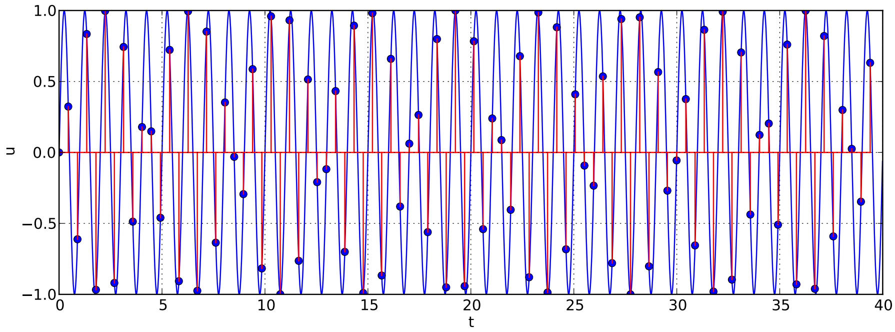
Pour illuster le théorème de Shannon, considérons tout d'abord le cas d'une fonction sinusoïdale. On définit une fonction de période 1 :

```
import math
def u(t):
    return math.sin(2*math.pi*t)
```

La fréquence maximale est évidemment $f_{max} = 1$. Nous allons effectuer deux échantillonnages de cette fonction. Le premier avec une fréquence grande devant 1, pour tracer la sinusoïde, le second avec une fréquence plus faible mais respectant la condition de Nyquist-Shannon (supérieure à 2)

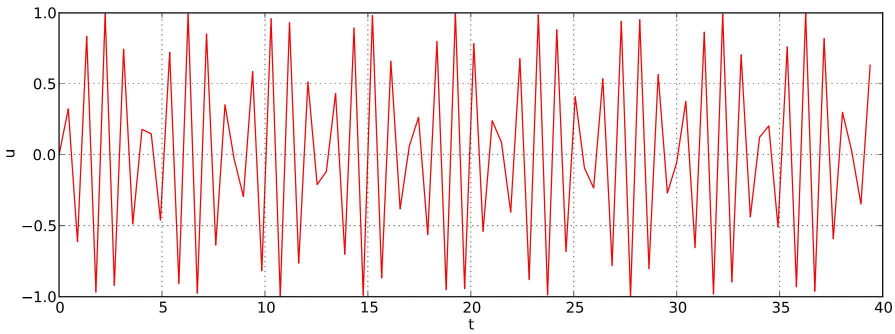
```
import numpy
from matplotlib.pyplot import *
T=40.0
fe1 = 100.0
te1 = 1.0/fe1
N1 = int(T*fe1)
t1 = numpy.arange(0,N1)*te1
x1 = numpy.zeros(N1)
for k in range(N1):
    x1[k] = u(t1[k])
fe2 = 2.234
te2 = 1.0/fe2
N2 = int(T*fe2)
t2 = numpy.arange(0,N2)*te2
x2 = numpy.zeros(N2)
for k in range(N2):
    x2[k] = u(t2[k])
figure(figsize=(12,4))
```

```
plot(t1,x1,"b")
stem(t2,x2,"r")
xlabel("t")
ylabel("u")
grid()
```



Si l'on relie les échantillons par des segments, on obtient bien sûr une très mauvaise représentation de la sinusoïde :

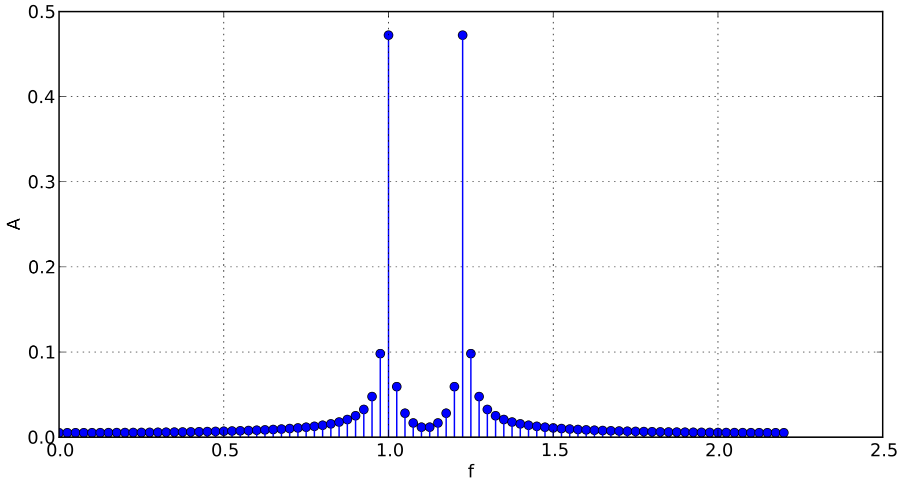
```
figure(figsize=(12,4))
plot(t2,x2,"r")
xlabel("t")
ylabel("u")
grid()
```



D'après le théorème de Shannon, il est pourtant possible de reconstruire complètement le signal. Pour le faire, nous allons plutôt nous placer dans l'espace des fréquences, en calculant la transformée de Fourier discrète des échantillons.

```
import numpy.fft
tfd = numpy.fft.fft(x2)
f = numpy.arange(0,N2)*1.0/T
figure(figsize=(10,5))
stem(f,numpy.absolute(tfd)/N2)
xlabel('f')
ylabel('A')
grid()
```

```
print(N2)
--> 89
```

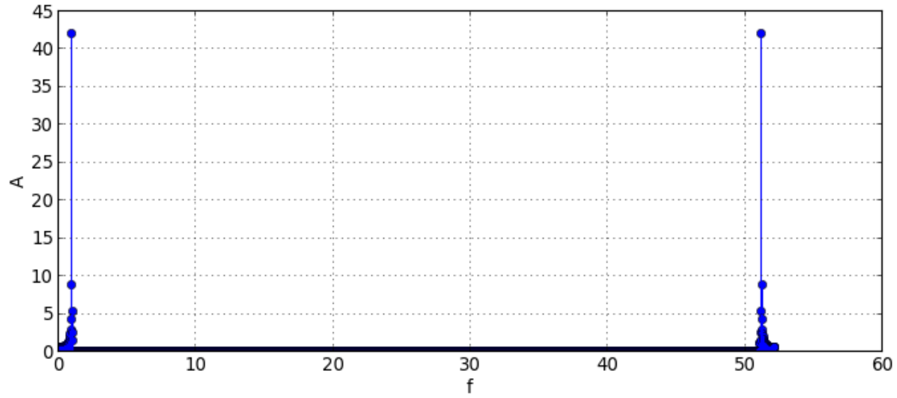


Le spectre du signal discret comporte deux maxima, le premier à la fréquence 1, et son image à la fréquence $2.234 - 1$. Nous allons couper la transformée de Fourier discrète en deux parties :

```
tfd_A = tfd[0:N2/2]
tfd_B = tfd[N2/2:N2]
```

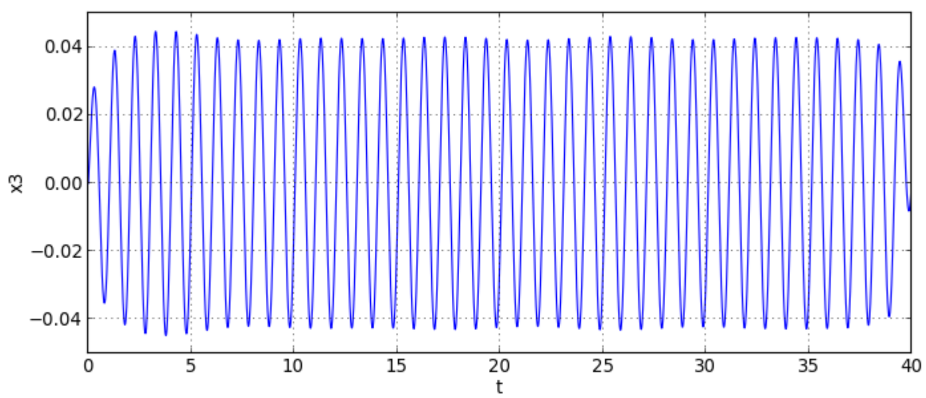
On peut à présent augmenter la fréquence d'échantillonnage en ajoutant des zéros entre ces deux parties. L'intervalle de fréquence entre deux points voisins reste $1/T$. La nouvelle fréquence d'échantillonnage se calcule à partir du nombre de points total.

```
nz = 2000
N3 = nz+N2
zeros = numpy.zeros(nz)
tfd3 = numpy.concatenate((tfd_A,zeros,tfd_B))
fe3 = N3/T
f = numpy.arange(0,N3)*1.0/T
figure(figsize=(10,4))
stem(f,numpy.absolute(tfd3))
xlabel('f')
ylabel('A')
grid()
```



Cette opération effectuée dans le domaine fréquentiel revient à augmenter la fréquence d'échantillonnage sans perdre d'information. Il reste à effectuer la transformée de Fourier discrète inverse :

```
x3 = numpy.fft.ifft(tfd3)
t3 = numpy.arange(0,N3)*1.0/fe3
figure(figsize=(10,4))
plot(t3,x3)
xlabel('t')
ylabel('x3')
grid()
```



Bien que le résultat ne soit pas parfait, nous obtenons une reconstruction de la sinusoïde initiale. La formule de Shannon (4) s'applique à un signal non limité dans le temps. En pratique, il a une durée finie T, c'est pourquoi la reconstruction est imparfaite. Nous verrons plus loin que la reconstruction se fait en pratique dans le domaine temporel et non pas de cette manière.

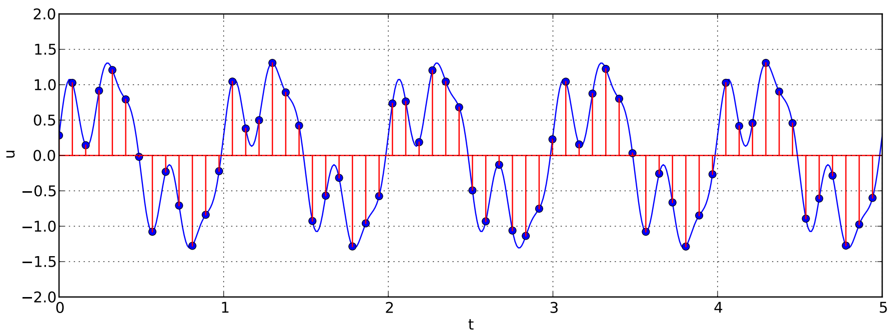
2.c. Signal périodique

Une fonction périodique est décomposée en somme de fonctions sinusoïdales (série de Fourier). Voici un exemple avec 3 harmoniques, de rang 1,3 et 5 :

```
def u(t):
    return 1.0*math.sin(2*math.pi*t)\
          +0.5*math.sin(3*2*math.pi*t+math.pi/3)\
          +0.3*math.sin(5*2*math.pi*t-math.pi/6)
```

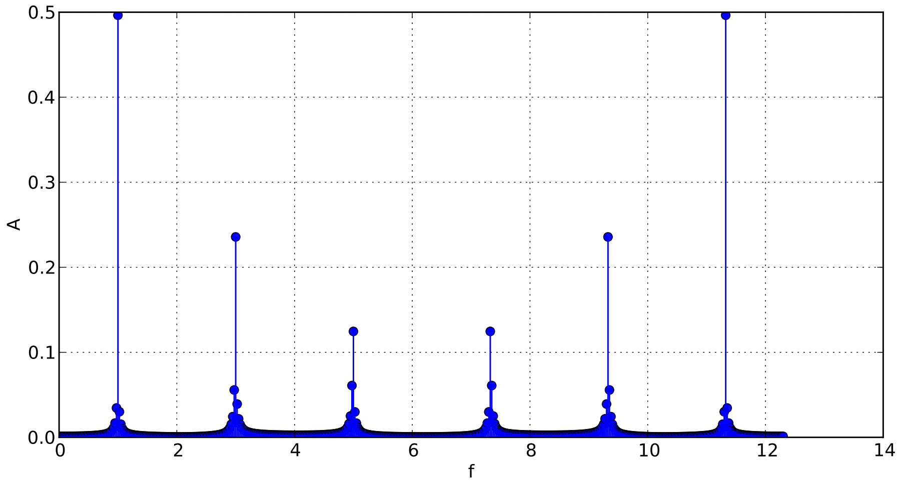
La plus grande fréquence du spectre du signal est celle de l'harmonique de rang 5. Pour respecter la condition de Nyquist-Shannon, il faut donc une fréquence d'échantillonnage supérieure à 10. On échantillonne 40 périodes avec une fréquence 12.345. La fréquence d'échantillonnage est choisie non multiple de celle du signal, comme c'est le plus souvent en réalité.

```
T=40.0
fe1 = 100
te1 = 1.0/fe1
N1 = int(T*fe1)
t1 = numpy.arange(0,N1)*te1
x1 = numpy.zeros(N1)
for k in range(N1):
    x1[k] = u(t1[k])
fe2 = 12.345
te2 = 1.0/fe2
N2 = int(T*fe2)
t2 = numpy.arange(0,N2)*te2
x2 = numpy.zeros(N2)
for k in range(N2):
    x2[k] = u(t2[k])
figure(figsize=(12,4))
plot(t1,x1,"b")
stem(t2,x2,"r")
xlabel("t")
ylabel("u")
axis([0,5,-2,2])
grid()
```



Voyons le spectre du signal discret :

```
tfd = numpy.fft.fft(x2)
f = numpy.arange(0,N2)*1.0/T
figure(figsize=(10,5))
stem(f,numpy.absolute(tfd)/N2)
xlabel('f')
ylabel('A')
grid()
```

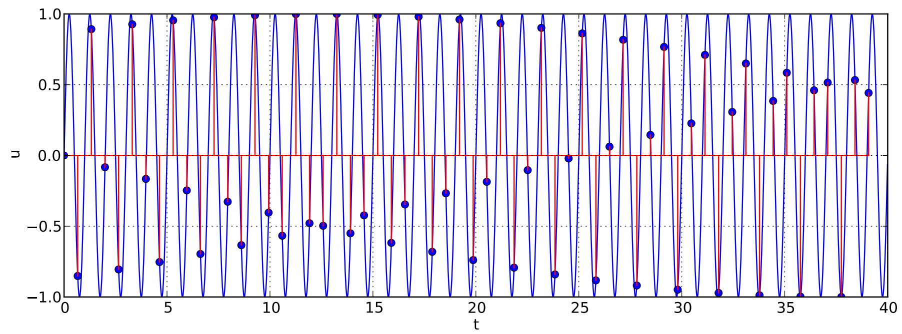


On voit sur ce spectre que la condition de Nyquist-Shannon est bien respectée : le spectre du signal continu, constitué des trois raies de fréquences 1,3 et 5, est bien obtenu sur la première moitié. Autrement dit, le spectre du signal et son image ne se chevauchent pas. Comme pour la sinusoïde, il est possible de reconstituer complètement le signal à partir des échantillons.

2.d. Repliement de bande

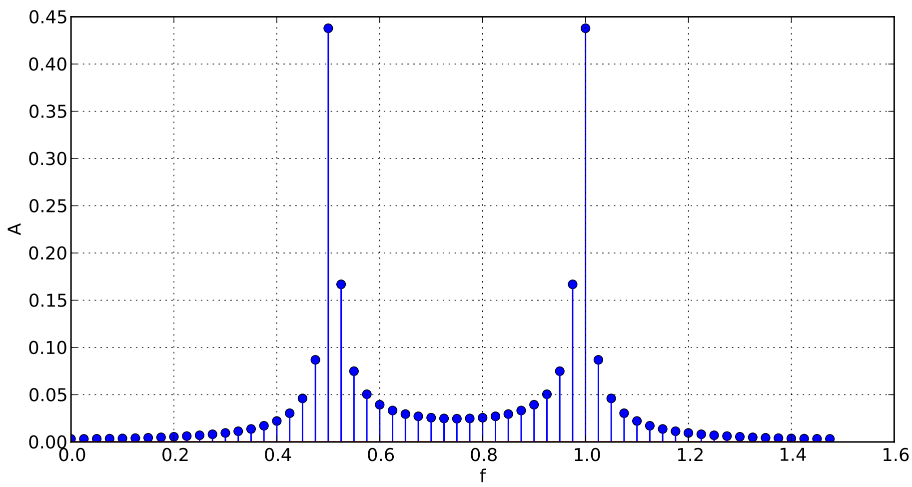
Le repliement de bande se produit lorsque la condition de Nyquist-Shannon n'est pas respectée. Voici un exemple de sinusoïde sous-échantillonnée :

```
def u(t):
    return math.sin(2*math.pi*t)
T=40.0
fe1 = 100.0
te1 = 1.0/fe1
N1 = int(T*fe1)
t1 = numpy.arange(0,N1)*te1
x1 = numpy.zeros(N1)
for k in range(N1):
    x1[k] = u(t1[k])
fe2 = 1.51
te2 = 1.0/fe2
N2 = int(T*fe2)
t2 = numpy.arange(0,N2)*te2
x2 = numpy.zeros(N2)
for k in range(N2):
    x2[k] = u(t2[k])
figure(figsize=(12,4))
plot(t1,x1,"b")
stem(t2,x2,"r")
xlabel("t")
ylabel("u")
grid()
```



et le spectre du signal discret :

```
tfd = numpy.fft.fft(x2)
f = numpy.arange(0,N2)*1.0/T
figure(figsize=(10,5))
stem(f,numpy.absolute(tfd)/N2)
xlabel('f')
ylabel('A')
grid()
```



Le spectre obtenu est toujours symétrique par rapport à la fréquence de Nyquist, mais la partie de gauche ne correspond pas du tout au spectre du signal continu, puisque le maximum se trouve à 0.5 au lieu de 1. Ce spectre comporte en fait une raie à la fréquence $f = 1$ du signal et une autre à la fréquence $f_e - f$. Si on cherche à reconstituer le signal continu à partir de ces échantillons, on obtient une sinusoïde de fréquence $f_e - f = 0.51$, de plus basse fréquence que la sinusoïde initiale. L'apparition de basses fréquences parasites est une conséquence du sous-échantillonnage qui peut être très gênante. Non seulement il y a perte d'information, mais il apparaît des informations non présentes dans le signal continu d'origine.

Le spectre obtenu peut s'interpréter en remarquant que le spectre d'une sinusoïde échantillonnée comporte toujours deux raies de fréquences f et $f_e - f$. Lorsque la condition de Nyquist-Shannon est respectée $f < f_e - f$. Lorsqu'elle ne l'est pas $f_e - f < f$. On dit que l'image du spectre (la raie $f_e - f$) s'est repliée dans la bande de fréquence $[0, f_e/2]$, c'est pourquoi on parle de repliement de bande.