MODÉLISATION MATHÉMATIQUE

Séances 5-6

THÈME : Séries de Fourier et traitement de signal

21 février 2013

Table des matières

1 Introduction

Ces deux séances seront consacrées à l'étude des applications pratiques des séries de Fourier à l'étude des signaux.

2 Séance 5

Problème 1. Comment calculer les coefficients de Fourier ? Nous allons considérer ici une fonction périodique de période 1 $f(t)$. On supposera que f est de classe $C^p[0,1]$ avec $p \geq 2$. On va travailler avec la forme complexe de la série de Fourier de f

$$\sum_{n=-\infty}^{+\infty} c(n)e^{2\pi inx}$$

où

$$c(n) = \int_0^1 f(t)e^{-2\pi int}dt$$

1. Soit une fonction
$$f(x) = e^{-x^2}$$
sur l'intervalle $[0,1]$. Pouvez vous calculer les coefficients de Fourier ? Si oui, comment ? Si non, pourquoi ?
2. Pour calculer de façon approcher les coefficients de Fourier d'une fonction on propose le procédé suivant. On choisit un par Δt et on définit une subdivision régulière de l'intervalle $[0,1]$
$$0 = t_0 < t_1 < \cdots < t_M = 1, \ \ t_k = k * \Delta t = \frac{k}{M}$$
et on pose
$$\widetilde{c_M(n)} = \frac{1}{M}\sum_{k=0}^{M-1} f\left(\frac{k}{M}\right)e^{-2\pi in\frac{k}{M}}$$
Expliquer pourquoi quand $M \to \infty$ on a $\widetilde{c_M(n)} \to c(n)$.

2.1 Transformée de Fourier discrète et techniques de compression

Problème 2. En utilisant les résultats de l'exercice précédent on peut définir une transformation qui est une approximation de la série de Fourier pour une fonction. Soit $N \in \mathbb{N}$. On considère la somme partielle d'ordre N de la série de Fourier

$$S_N(x) = \sum_{n=-N}^{N} c(n)e^{2\pi inx}$$

où

$$c(n) = \int_0^1 f(t)e^{-2\pi int}dt$$

Ensuite on choisit de subdiviser l'intervalle $[0, 1]$ en précisément $M = 2N$ intervalles pour le calcul approché des coefficients. On obtient ainsi les $2N$ coefficients de Fourier approchés :

$$c_{2N}(n) = \frac{1}{2N} \sum_{k=0}^{2N-1} f\left(\frac{k}{2N}\right) e^{-2\pi i n \frac{k}{2N}}$$

Cette construction permet de retrouver précisément les valeurs de la fonction f aux points de la subdivision $t_k = \dfrac{k}{2N}$ par l'égalité

$$f\left(\frac{k}{2N}\right) = \sum_{k=-N}^{N} c_{2N}(n) e^{2\pi i n \frac{k}{2N}}$$

Nous allons d'abord le constater par expérience.

1. Soit une fonction

   $$f(x) = x(1 - x)$$

   sur l'intervalle $[0, 1]$. On utilisera la commande `dft` de scilab pour calculer les coefficients de Fourier.

```
fGraph=zeros(1,2*N+1);
for k=1:2*N+1
  fGraph(k)=f(t(k));
end
xset("window", 2);
plot2d(t,fGraph);// représentation graphique de la fonction

Cn=dft(fGraph,-1)// transformée de Fourier discrete

xset("window", 3);
plot2d(0:2*N,abs(Cn));// représentation graphique des coefficients de Fourie

fn=dft(Cn,1)// transformée de Fourier discrete inverse



xset("window", 4);
plot2d(t,fn);// représentation graphique des coefficients de Fourier
```

2. On rappelle ici la propriété importante des coefficients de Fourier d'une fonction de classe C^p. Il existe une constante C telle que pour tout $n \in \mathbb{Z}$

   $$|c(n)| \leq \frac{C}{|n|^p}$$

   On dispose d'un enregistrement numérique d'un signal comportant $2 * N + 1$ échantillons. Comment peut on approcher ce signal par un autre qui prend moins de place (moins d'échantillons) ?

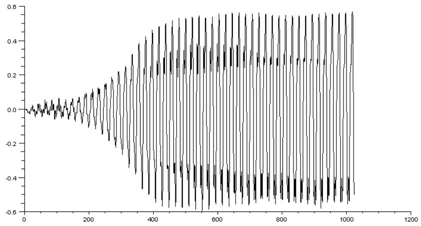

FIGURE 1 – son de flute

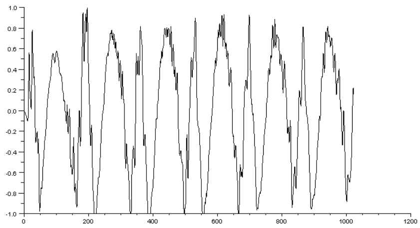

FIGURE 2 – son de bass

2.2 Transformée de Fourier discrète et transformation des sons

Les sons que nous entendons sont des ondes qui se propagent dans l'air, par exemple, par les oscillations de la pression. Ce sont ces oscillations qui sont captées par l'oreille et transformées ensuite en signaux électriques arrivant jusqu'au cerveau. Ce sont les fréquences de ces oscillations qui déterminent notre perception des sons et nous permettent de les distinguer entre eux. Voici par exemple, deux ondes correspondantes aux sons de flute et de bass :

On peut considérer un signal comme une fonction de temps, $s(t)$. Si la durée du signal est finie, T on peut le considérer comme une fonction périodique de période T. Alors la série de Fourier d'une telle fonction peut être interprétée comme la décomposition du signal en somme de signaux particuliers,

$$\cos\left(\frac{2\pi n}{T}t\right) \text{ et } \sin\left(\frac{2\pi n}{T}t\right)$$

Dans le cas de signal sonore il s'agit de représenter une onde complexe, de fréquence $\dfrac{1}{T}$ comme somme d'ondes simples, sinusoïdales, dont les fréquences sont des multiples de $\dfrac{1}{T}$.

C'est cette décomposition en fréquences pures qui distingue un son de l'autre. On appelle l'ensemble de coefficients de Fourier d'un signal le **spectre de fréquences**. La représentation graphique de la suite de

$$|C_n| = \sqrt{a_n^2 + b_n^2}$$

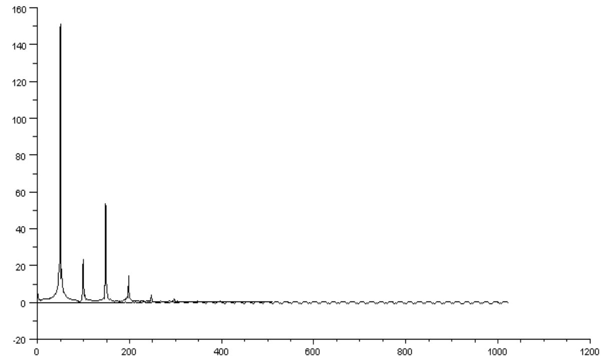

FIGURE 3 – son de flute

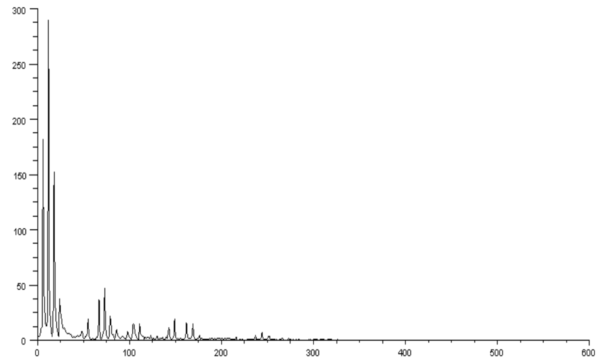

FIGURE 4 – son de bass

s'appelle le spectre d'amplitude. Elle permet de visualiser la composition d'un son en fréquences pures et étudier ses propriétés.

Par exemple, pour les deux signaux que nous avons cités ci-dessus voici les spectres :

La fréquence $\frac{1}{T}$ s'appelle la fondamentale. La grandeur des coefficients dans la décomposition reflète l'importance relative des fréquences correspondantes. Si la fréquence est petite, on dis que le son est bas. Si elle est grande on dit qu'il est haut. On peut ainsi "voir" les caractéristiques d'un son en observant son spectre de fréquences. On remarquera alors que le son de bass est "bas" car composé essentiellement des fréquences basses et que le son de flûte est plus haut.

Problème 3. Synthèse de son

A l'aide de scilab on peut extraire des informations d'un fichier sonore au format WAV et enregistrer un tel fichier. Voici les commandes :

```
y=wavread('MonJoliSon.wav',[1 M] );// lit dans le fichier indiqué les M prem
 wavwrite(y,'toto.wav');//enregistre le tableau y comme fichier wav
```

1. Vous trouverez sur AREL un fichier Scilab et deux fichiers son. Il s'agit d'enregistrements d'un même mot, pomme, prononcé par deux voix différentes. Le fichier scilab permet de lire les données des deux sons, les afficher, calculer les spectres et afficher également les spectres. exécutez le fichier et dites lequel des deux sons est plus grave. Ecoutez les et vérifiez votre analyse.
2. Mixez les deux sons, en calculant leur spectre moyen. Enrestrez. Ecoutez.
3. Amusez vous à transformer le spectre et observer le résultat sur le son.

3 Séance 6. Transformée de Fourier et les fréquences

Problème 4. Compression à l'aide de la transformée de Fourier

Lors de la séance de préparation nous avons discuté des possibilités de compresser un signal à l'aide de la transformée de Fourier. Il s'agit maintenant de mettre en pratique les idées qui ont été discutées.

A l'aide de scilab on peut extraire des informations d'un fichier sonore au format WAV et enregistrer un tel fichier. Voici les commandes :

```
y=wavread('MonJoliSon.wav',[1 M] );// lit dans le fichier indiqué les M prem
 wavwrite(y,'toto.wav');//enregistre le tableau y comme fichier wav
```

Vous trouverez sur AREL un fichier Scilab et deux fichiers son. Il s'agit d'enregistrements d'un même mot, pomme, prononcé par deux voix différentes. Le fichier scilab permet de lire les données des deux sons, les afficher, calculer les spectres et afficher également les spectres.

On simulera la compression en remplaçant certains coefficients de Fourier par des zéros. Supposons que le signal comporte M échantillons. On souhaite en éliminer L. On définit alors le taux de compression comme le rapport entre la longueur du signal compressé et celle du signal original :

$$t = \frac{M - L}{M}$$

L'objectif et de tester et de comparer deux approches de compression :

1. Suppression des L derniers coefficients de Fourier
2. Suppression des L plus petits coefficients de Fourier.

Travail à faire.

- Ecrire une fonction qui compresse le signal selon la première méthode. Cette fonction prends en paramètres
  - le nom du fichier contenant un son,
  - la longueur du signal à extraire, M,

  - le nombre de coefficients de Fourier à supprimer, L
  - le nom du fichier de sortie

  . Elle effectue les opérations suivantes :
  - Extraire M échantillons du signal à partir du ficheir, $s(n)$
  - Calculer les coefficients de Fourier (utiliser pour cela la fonction correspondante de Scilab)
  - Dans le tableau des coefficients de Fourier annuler les L derniers
  - Reconstruire le signal à partir des coefficients de Fournier modifiés $s1(n)$
  - Enregistrer le son dans un fichier WAV
  - Calculer et renvoyer l'erreur quadratique :

$$E = \sqrt{\sum_{n=0}^{M-1} (s(n) - s1(n))^2}$$

- Ecrire une fonction qui compresse le signal selon la deuxième méthode. Cette fonction prends en paramètres
  - le nom du fichier contenant un son,
  - la longueur du signal à extraire, M,
  - le nombre de coefficients de Fourier à supprimer, L
  - le nom du fichier de sortie

  . Elle effectue les opérations suivantes :
  - Extraire M échantillons du signal à partir du ficheir, $s(n)$
  - Calculer les coefficients de Fourier (utiliser pour cela la fonction correspondante de Scilab)
  - Dans le tableau des coefficients de Fourier identifier les L plus petits coefficients, puis les annuler
  - Reconstruire le signal à partir des coefficients de Fournier modifiés $s1(n)$
  - Enregistrer le son dans un fichier WAV
  - Calculer et renvoyer l'erreur quadratique :

$$E = \sqrt{\sum_{n=0}^{M-1} (s(n) - s1(n))^2}$$

- Effectuer plusieurs tests en faisant varier M et L et tracer les graphes de l'erreur E en fonction du taux de compression t pour chacune des méthodes de compression.
- Analyser les résultats et proposer les explications, en utilisant la théorie