

TP2

Echantillonnage des signaux

Cas de la parole et un signal test

I. Objectif du TP

Le but du TP est de comprendre le concept de numérisation des signaux en se basant sur la pratique avant d'étudier la théorie en classe. En effet, avec cette approche anticipative, l'élève ingénieur sera sensibilisé au théorème d'échantillonnage de manière expérimentale sur des signaux réels. Le cours, à travers les développements analytiques dans le domaine de Fourier, confirmera les résultats de l'expérience.

Durant ce TP, le principe de l'échantillonnage sera illustré avec un signal de parole généré par l'élève ingénieur et acquis avec la carte son du PC ainsi que sur un signal ECG déjà découvert durant le premier TP. Ainsi, en manipulant plusieurs fréquences d'échantillonnage, l'élève ingénieur aura une meilleure appréciation des conditions d'échantillonnage d'un signal.

Pour atteindre ces objectifs, les manipulations suivantes seront effectuées sous Matlab.

Signal de parole : enregistrer une séquence de parole en spécifiant la fréquence d'échantillonnage et le nombre de bits par échantillon, sauvegarder la séquence sous forme de fichier numérique sous format WAV, tracer et écouter le signal temporel, calculer et tracer sa représentation fréquentielle, modifier la fréquence d'échantillonnage, voir son impact sur la qualité auditive du signal de parole et conclure sur la fréquentielle d'échantillonnage assurant une excellente restitution auditive.

Signal ECG : re-découvrir la condition d'échantillonnage (théorème de Shannon) par l'expérience sur un signal ECG en modifiant la fréquence d'échantillonnage, en observant l'allure temporelle, la représentation spectrale et en comparant par rapport à celle du signal original fourni.

II. Principe de la numérisation d'un signal de parole

Une chaîne d'acquisition et de traitement numérique d'un signal de parole est illustrée sur la Figure 1. Ses étapes sont les suivantes :

- Transformation du signal analogique à traiter en un signal électrique proportionnel : c'est la capture ou l'acquisition du signal. Elle se fait à travers un microphone dans le cas de la parole qui sera un des signaux d'illustration durant ce TP.
- Conversion du signal électrique en une suite de valeurs numériques binaires, seules compréhensibles par les calculateurs numériques. C'est la conversion analogique-numérique.
- Exécution du traitement voulu, comme par exemple la compression ou la transmission.
- Conversion des codes binaires résultant du traitement en un signal électrique pour ramener le résultat final dans le monde réel analogique. C'est la conversion numérique-analogique.
- Reconversion du signal électrique dans la grandeur physique initiale, à travers un système d'écoute de type haut parleurs dans le cas d'un signal de parole par exemple.

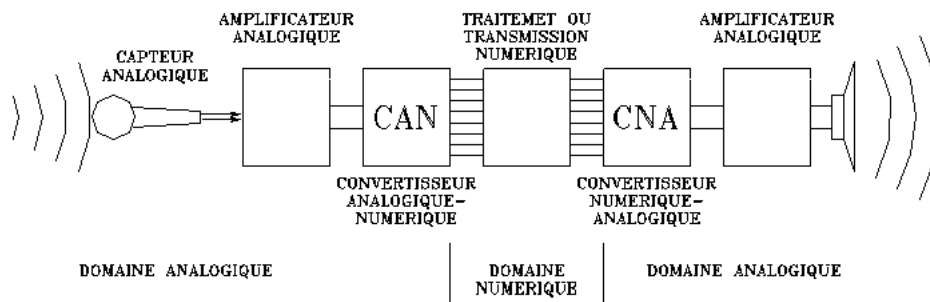


Figure 1 : Système de traitement analogique/numérique/analogique de la parole.

Dans ce qui suit, nous allons nous intéresser à la numérisation du signal de parole. Un signal analogique est un signal continu qui peut prendre une infinité de valeurs possibles alors qu'un signal numérique est un signal discret (discontinu), qui se résume en une succession de « 0 » et de « 1 ». L'objectif de la numérisation est de transformer le signal analogique qui contient une quantité infinie d'amplitudes en un signal numérique contenant une quantité finie de valeurs. Le passage de l'analogique au numérique se fait en 2 étapes successives : l'échantillonnage et la quantification.

L'échantillonnage est la première étape de la numérisation des signaux. Elle permet de convertir un signal continu dans le temps (Figure 2.a) en un signal discret dans le temps (Figure 2.b). Elle se fait par prélèvement discontinu régulier de "tranches temporelles" ou échantillons à une certaine cadence. Le nombre d'échantillons prélevés par secondes s'appelle la fréquence d'échantillonnage exprimée en HERTZ. Le nombre d'échantillons composant le signal numérique devrait être suffisamment grand pour pouvoir représenter le signal analogique de départ mais pas trop grand non plus pour ne pas être trop volumineux. C'est pour cela qu'il est important de savoir choisir la fréquence d'échantillonnage, ce qui constitue le premier objectif d'apprentissage de ce TP.

La Quantification Scalaire (QS) est la deuxième étape de la numérisation. Elle permet de transformer le signal discret en temps mais continu en amplitudes en un signal discret en amplitude (Figure 2.c). La résolution numérique du signal est le nombre de « niveaux » ou de « paliers » qu'il est possible d'enregistrer pour reproduire l'amplitude du signal. Avec une résolution de 8 bits par exemple, on dispose de 2^8 , soit 256 valeurs possibles pour traduire l'amplitude du son. Plus la résolution est élevée, meilleure sera la représentation discrète du signal.

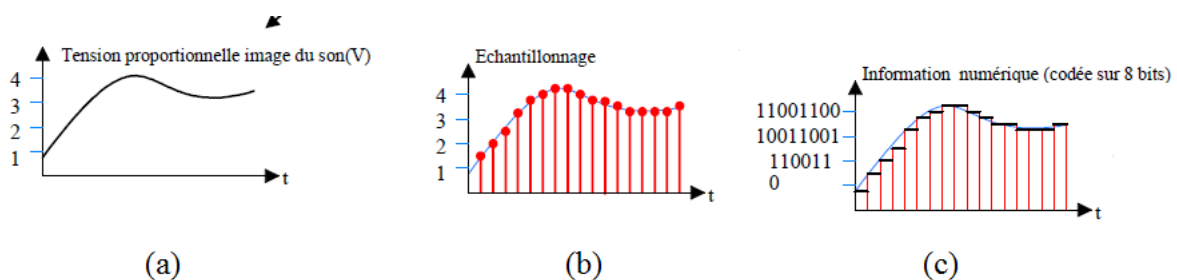


Figure 2 : Illustration des étapes de la numérisation.

Les signaux de parole manipulés durant ce TP auront le format WAV (Waveform). C'est un dérivé de la spécification RIFF (Resource Interchange File Format) de Microsoft dédiée au stockage de données multimédias. Ce format est libre d'utilisation et est parmi les plus répandus de formats de fichiers sons. Le format WAV considère les paramètres de numérisation suivants : la fréquence d'échantillonnage, la résolution de l'échantillon et le nombre de canaux d'enregistrement : 1 (mono), 2 (stéréo).

III. Recommandations générales pour le TP

S'il est possible, on vous demande de ramener un kit microphone/haut parleur. Ainsi, vous pouvez enregistrer et écouter les sons convenablement et éviter les interférences avec les autres.

Notations :

- x : Séquence (vecteur) d'échantillons.
- fs : Fréquence d'échantillonnage.
- $Nbits$: Nombre de bits par échantillon.
- $Path$: Répertoire courant (où se trouve la donnée ou le code).
- $N_duration$: Nombre d'échantillons à acquérir, qui est le produit de la durée en secondes N_s et la fréquence d'échantillonnage : $N_duration = N_s * fs$.

Fonctions :

- Lire un fichier audio en format WAV : `[x, fs] = wavread('Path\Name.wav')`.
- Ecrire un fichier audio en format WAV : `wavwrite(x, fs, Nbits, 'Path\Name.wav')`.
- Ecouter une séquence audio : `soundsc(x, fs)` ou `sound(x, fs)`.
- Enregistrer un fichier audio avec la carte son du PC. Deux propositions sont fournies : `wavwrite(x, fs, Nbits, 'Path\Name.wav')` ou `x = wavrecord(N_duration, fs, type_data, Ch)`.

IV. Acquisition d'un signal audio

Le programme MATLAB suivant permet d'acquérir un signal de parole à partir de la carte son du PC. Il est composé d'un programme principal et d'une fonction `My_Record`.

- a) Comprenez le programme principal et la fonction `My_Record`.
- b) Testez le fonctionnement de ce programme.
- c) Tracer l'allure temporelle du signal enregistré.
- d) Ecouter le signal.

```
fs = 16000; % Sampling frequency
Duration = 4; % Duration of speech (in seconds)
Nbits = 16; % Resolution of samples
Type_data = 'int16'; % Type of data during acquisition, can be 'double', 'single', 'int16', 'uint8', or 'int8'.
```

```
N_duration = Duration*fs;
```

```
[x] = My_Record(Duration, fs, Nbits, Type_data, 1); % Function to acquire speech
```

```
% Format conversion to double float
MySpeech = double(x);
```

```
% Signal normalization
MySpeech = MySpeech/max(abs(MySpeech));
```

```
function [MySpeech] = My_Record(Duration, fs, nBits, Type_data, nChannels)
```

```
    % Arguments
```

```
        % fs: Sampling frequency
```

```
        % nBits: Resolution: 8,16,24 bits
```

```
        % nChannels: 1: mono, 2: stereo
```

```
        % Duration: Duration in seconds
```

```
    % Fonctions
```

```
        % Display: Display message on the screen
```

```
        % Audiorecorder: Construction of audiorecorder class
```

```
        % Recordblocking: Maintaining recording during 'Duration'
```

```
    % Code
```

```

r = audiorecorder(fs, nBits, nChannels);
display('Begin recording');
% Speak into microphone ...
recordingblock(r, Duration);
disp('End recording');
MySpeech = getaudiodata(r, Type_data); % Get data as Type_data format
end

```

V. Modification de la fréquence d'échantillonnage

- Reprenez le programme précédent en variant la fréquence d'échantillonnage dans un ordre décroissant, de 9 kHz à 1 kHz, avec un pas de 2 kHz. Pour cela, il est recommandé d'utiliser une boucle dans le code MATLAB, en introduisant une pause manuelle entre les itérations (commande MATLAB `pause`).
- Prononcez à chaque fois la même phrase.
- Évaluez la qualité auditive de la séquence acquise en attribuant une note de 1 à 5 à la qualité pour chaque fréquence d'échantillonnage. On attribue les notes comme suit :

Note	5	4	3	2	1
Qualité	Excellente	Bonne	Dégradation faible	Dégradation remarquable	Dégradation extrême

- Que remarquez-vous au fur et à mesure que la fréquence d'échantillonnage diminue ? Que remarquez-vous pour les faibles fréquences d'échantillonnage particulièrement ?

VI. Analyse fréquentielle d'un signal ECG

Pour analyser les différentes représentations temporelles et fréquentielles pour des valeurs différentes de la fréquence d'échantillonnage d'un même signal analogique, on se propose d'étudier un signal ECG et de tracer son spectre d'amplitude pour différentes fréquences d'échantillonnage.

Le signal ECG à manipuler est disponible dans le fichier *ECG_10kHz.mat*. Il s'agit d'une version numérique acquise avec une fréquence d'échantillonnage relativement très grande et qui peut être assimilée à un celle d'un signal analogique. En effet, il est impossible de disposer d'une version analogique sur un ordinateur numérique, en l'occurrence l'ordinateur avec lequel nous travaillons.

VI.1 Méthodologie

Pour calculer et tracer le spectre d'amplitude du signal, il faut suivre les étapes suivantes.

- Calculer la Transformée de Fourier sur signal sur $N_{fft} = 1024$ points en utilisant la fonction *fft.m* de MATLAB.
- Tracer son module tout en normalisant (division par le maximum).
- L'axe des fréquences de la figure précédente est fourni selon le numéro de la raie de fréquence. Il n'est pas suffisamment significatif puisqu'il ne représente pas de vrais fréquences en Hz. De plus, le spectre d'amplitude obtenu n'est pas symétrique autour de zéro (tel est le cas pour les signaux réels). Pour améliorer la représentation spectrale, on suit les étapes suivantes :

- Créer un axe fréquentiel en Hz à N_{fft} points dans l'intervalle $[-f_s/2, f_s/2]$, avec f_s est la fréquence d'échantillonnage sélectionnée.

- Créer un nouveau vecteur formé par les $N_{fft}/2$ derniers points du spectre suivis des $N_{fft}/2$ premiers points du spectre.
- Tracer la nouvelle représentation fréquentielle en fonction du nouveau vecteur des fréquences en Hz.

VI.2 Travail demandé

- a) Tracer le signal test $x(n)$ échantillonné à la fréquence 10 kHz, qui sera désormais considéré comme étant le signal analogique $x(t)$.
- b) Tracer son spectre d'amplitude. Il sera considéré comme étant celui du signal analogique $x(t)$ et sera noté $|X(f)|$.
- c) Générer un signal discret obtenu en sous-échantillonnant le précédent signal par la valeur 2. L'opération de sous-échantillonnage consiste à retenir, dans l'ordre temporel, un échantillon sur 2. On obtient ainsi un nouveau signal qu'on note $x_{5kHz}(n)$.
- d) Calculer le spectre du signal test $x_{5kHz}(n)$ en réduisant le nombre de points de la *fft* par 2. Ce spectre est noté $|X_{5kHz}(f)|$.
 - i. Superposer sur le même graphe $|X(f)|$ et $|X_{5kHz}(f)|$. Pour des raisons de lisibilité, on se limitera à la bande fréquentielle s'étalant de -500 Hz à 500 Hz.
 - ii. Interpréter le résultat obtenu.

VII. Influence de la modification de la fréquence d'échantillonnage sur la représentation fréquentielle

VII.1 Analyse fréquentielle

- a) Reprendre la manipulation précédente en modifiant la fréquence d'échantillonnage à 312.5 Hz et en choisissant 32 points pour la *fft*.
- b) Comparer les représentations spectrales et conclure.
- c) Reprendre l'expérience pour des valeurs de la fréquence d'échantillonnage allant de 10 kHz à 39.0625 Hz en divisant à chaque fois la fréquence d'échantillonnage par 2 et le nombre de points de la *fft* par 2. Au démarrage, il faudra calculer la *fft* sur 1024 Points. On veillera à ce qu'une légende dynamique apparaisse sur la figure afin de faciliter l'interprétation des figures.
- d) A partir de quelle fréquence d'échantillonnage est-il possible d'avoir la même représentation fréquentielle que celle du signal à temps continu ?
- e) Conclure sur la condition d'échantillonnage.

VII.2 Analyse temporelle

- a) Superposer sur une même figure tous les signaux précédents dans le domaine temporel en utilisant de vrais axes de temps (et non des positions des échantillons).
- b) Comparer les allures temporelles et indiquer la fréquence d'échantillonnage à partir de laquelle des différences sont remarquables.
- c) Confronter ce résultat par rapport à celui de la représentation spectrale.