

Développement d'une application MATLAB permettant l'analyse et le traitement du signal audio

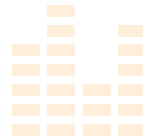
Lucie Perouas, étudiante L2 Sciences de l'ingénieur CUPGE

Encadrant : Gilles Chabriel, laboratoire IM2NP

Plan de la présentation



1. Contexte et objectifs
2. Architecture générale de l'application
3. Génération d'un signal sinusoïdal



4. Quantification

5. Analyse fréquentielle

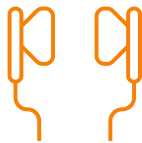


6. Import d'un signal audio

7. Sous-échantillonnage et repliement spectral

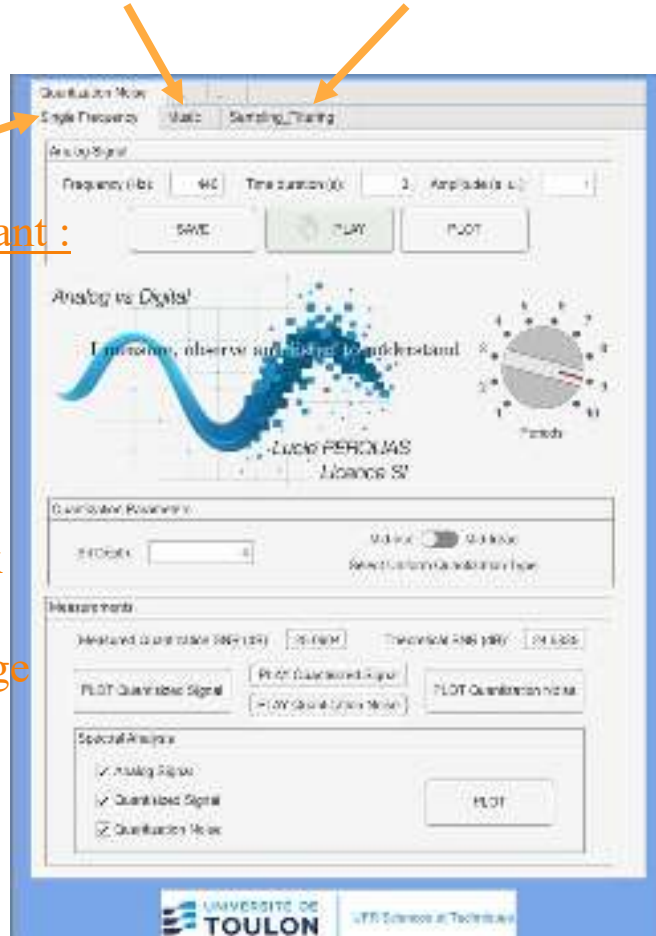
8. Filtrage numérique

9. Reconstruction du signal



10. Perspectives

Architecture générale de l'application

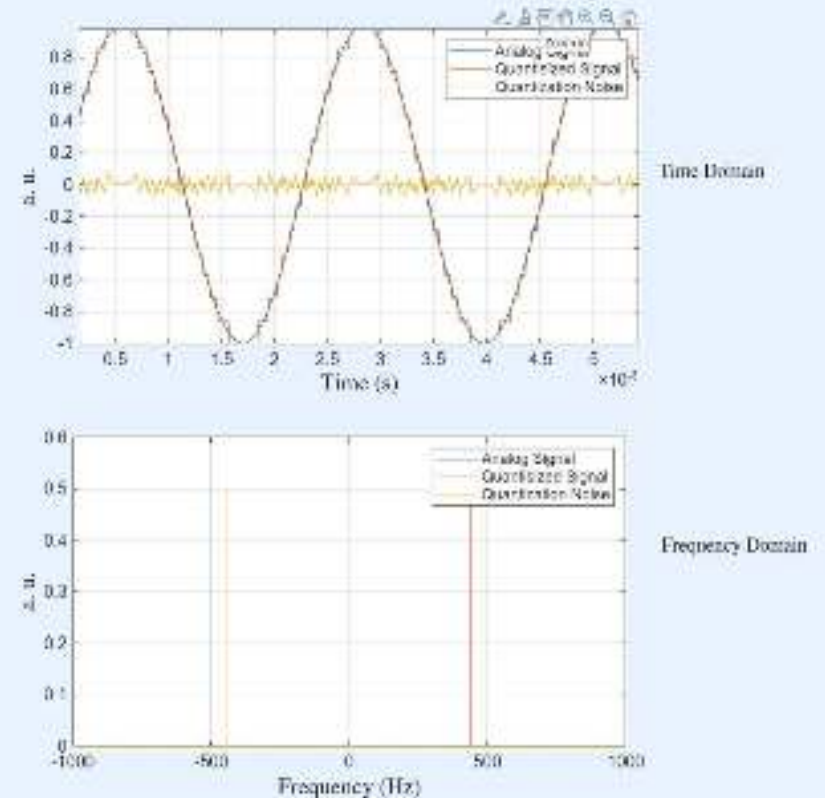


3 onglets permettant :

Traitement d'une monofréquence

Traitement de signaux musicaux

Ré-échantillonnage et filtrage



Génération d'un signal sinusoïdal

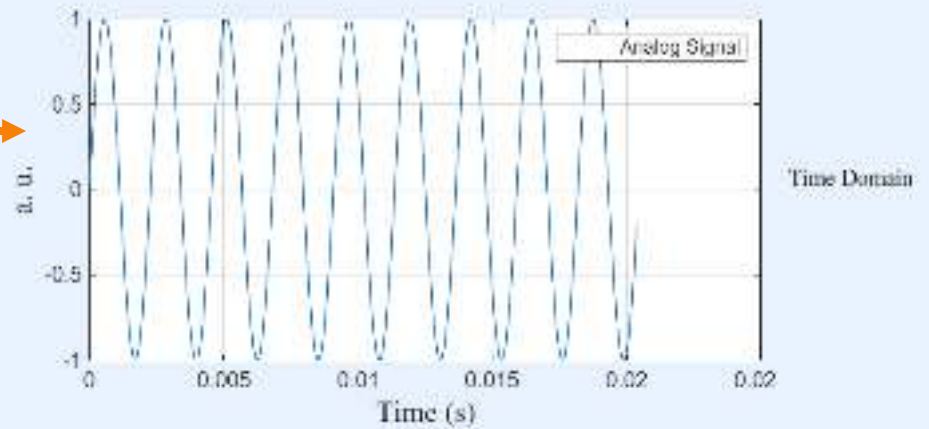
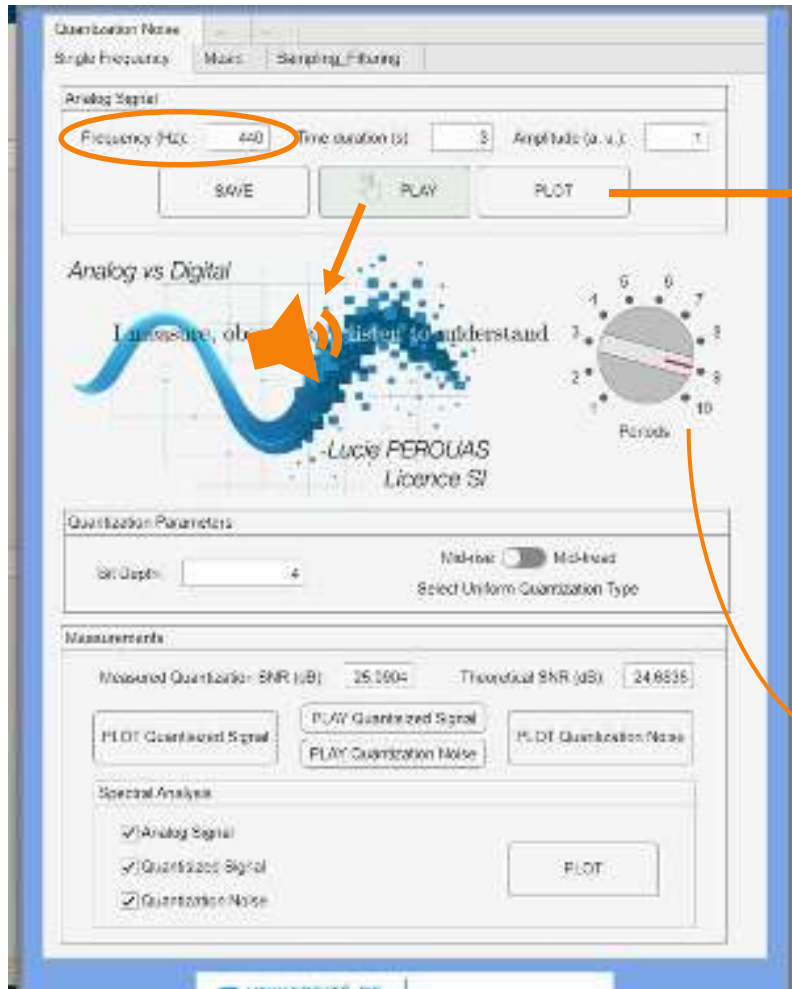
➤ Un signal sinusoïdal est de la forme : $x(t) = A \sin(2\pi f t + \varphi)$

➤ En version discrète cela donne $x[n] = A \sin(2\pi f n T_e + \varphi)$

➤ Par exemple : $x(t) = A \sin(2\pi \times 440 t)$

Le signal est échantillonné :
on n'a plus qu'un nombre
discret de valeurs

La fréquence est paramétrée par l'utilisateur dans
l'application



L'utilisateur voit sur le panneau de droite la représentation temporelle.

Il choisit avec la molette le nombre de périodes à visualiser.

Quantification

- Consiste à remplacer les amplitudes continues d'un signal par un nombre fini de niveaux, déterminé par le nombre de bits n .

```
function signalq = decreaseQuantization(signal,n,Ampl)
```

- Implémentation dans l'application :

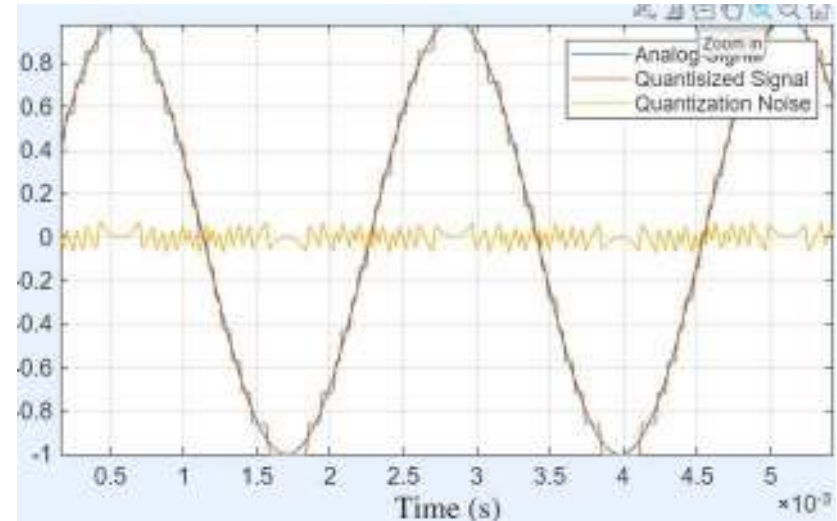
```
signalq = signal*(2^(n-1)-1)/Ampl;  
signalq = round(signalq); % mid-rise  
signalq = signalq/(2^(n-1)-1)*Ampl;
```

- Erreur de quantification :

$$e_q[n] = x_q[n] - x[n]$$

- Rapport signal à bruit :

$$\text{SNR}_{\text{dB}} = 10 \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{bruit}}} \right)$$



Analyse fréquentielle

- La transformée de Fourier du signal continu est :

$$X(f) = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi ft} dt$$

- Après échantillonnage, le signal devient une suite d'échantillons :

$$x[n] = x(nT_e), \quad T_e = \frac{1}{f_s}$$

- La TF est donc remplacée par la transformée de Fourier Discrète :

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}}, \quad k = 0, \dots, N-1$$

Configuration Menu

Signal Frequency: 1000 Hz, Amplitude: 1.0 V, Phase: 0°

Sampling Rate: 4000 Hz, Bits: 16, Format: Signed

Display: Waveform, Spectrum, Both

Time Domain

Frequency Domain

Quantization Parameters

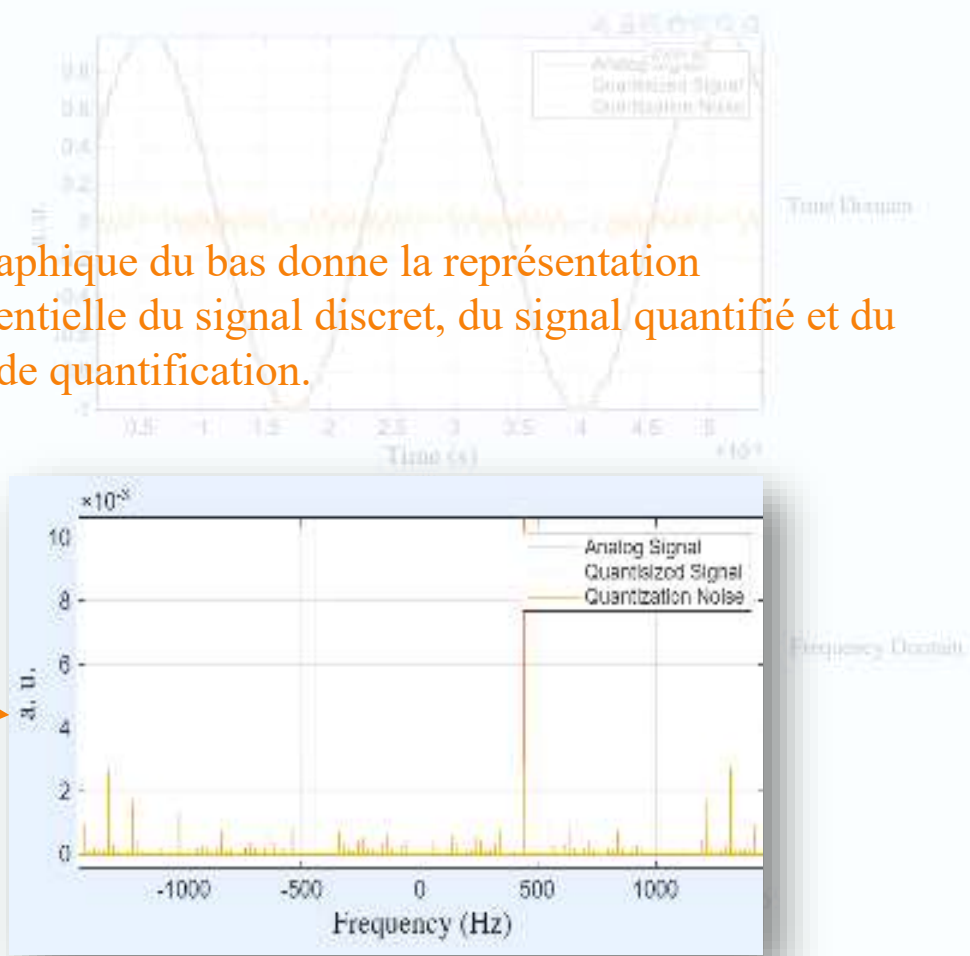
Bits: 16, Source: Analog, Quantization Type: Uniform

Measurement

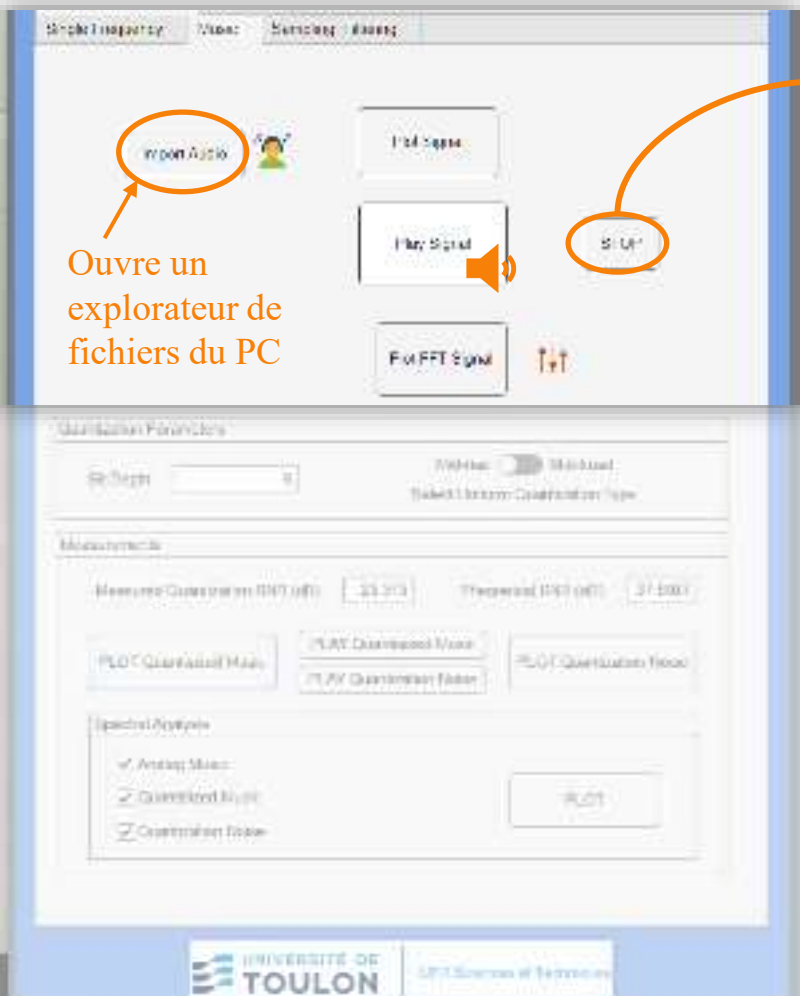
Waveform: Analog Signal, Quantized Signal, Quantization Noise

Buttons: Plot, Run, Stop, Help

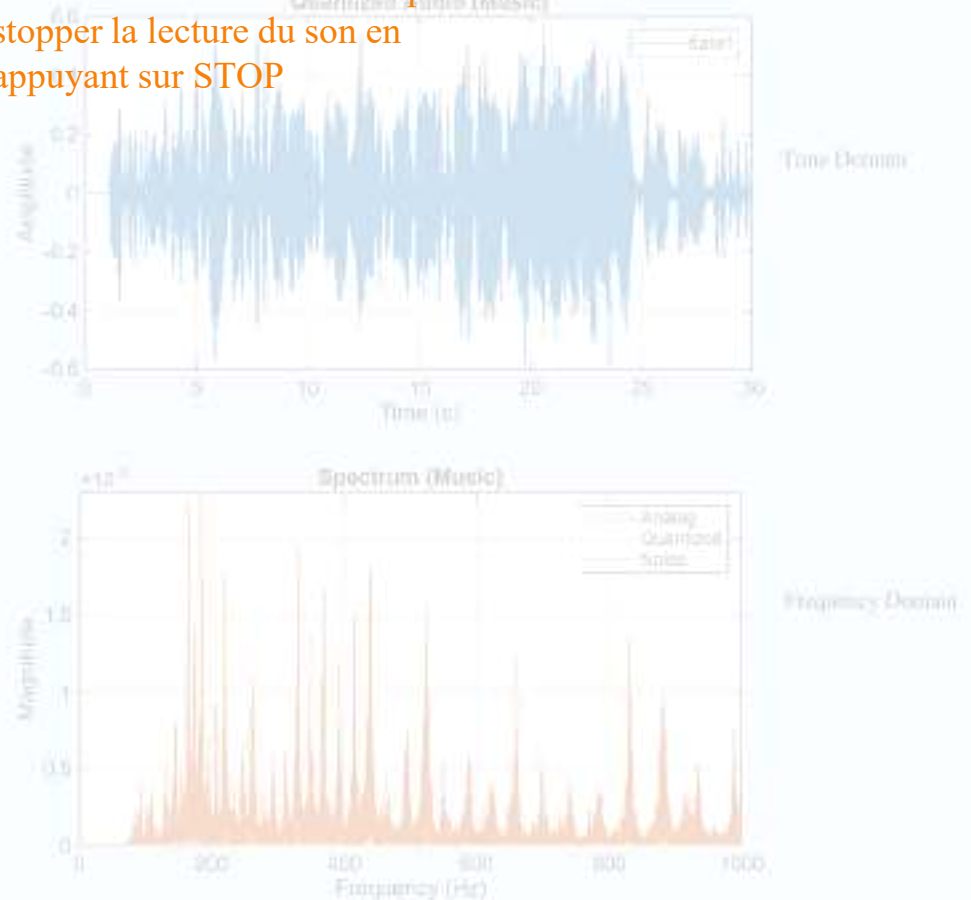
Le graphique du bas donne la représentation fréquentielle du signal discret, du signal quantifié et du bruit de quantification.



Import d'un signal Audio

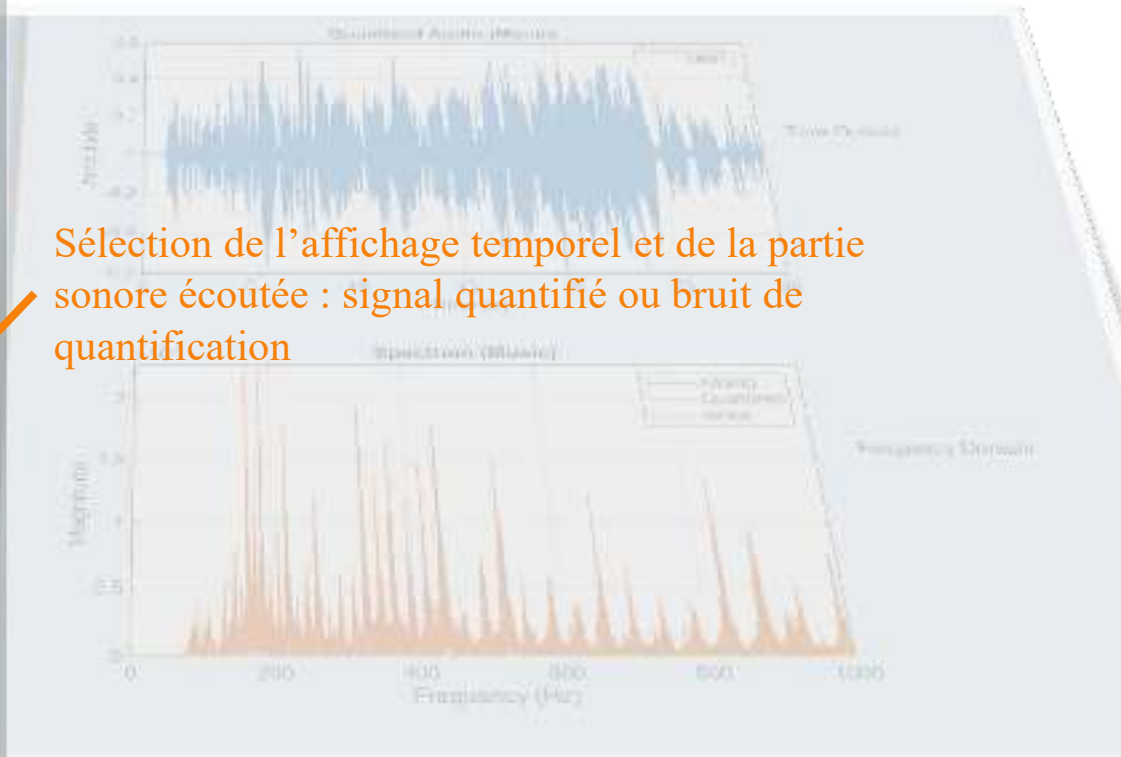
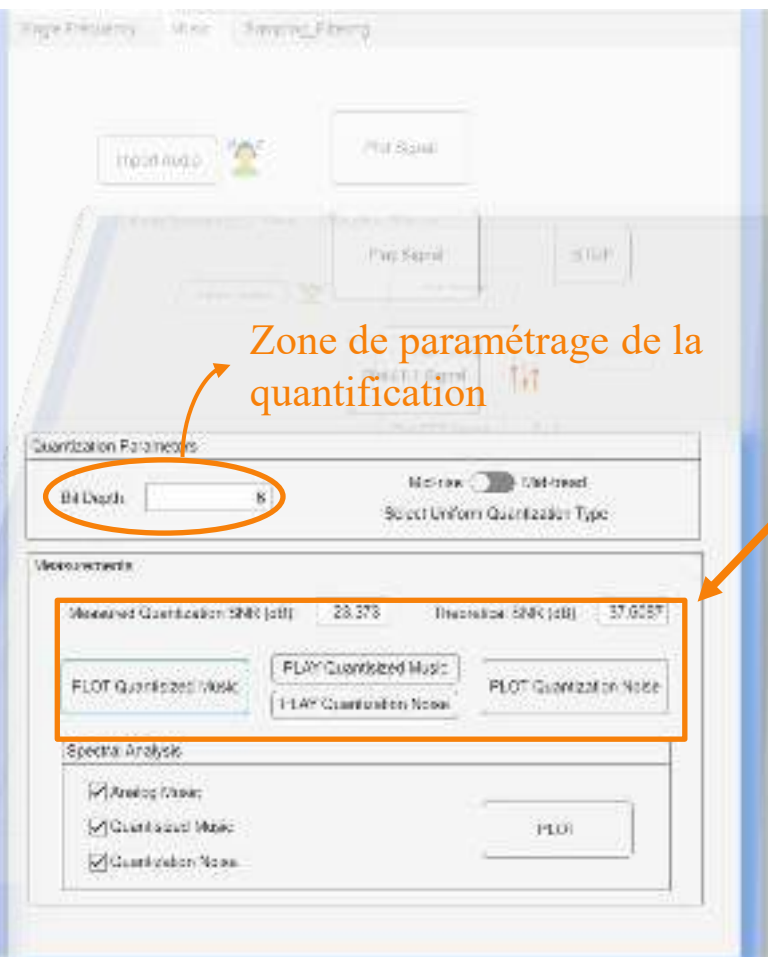


À tout moment l'utilisateur peut stopper la lecture du son en appuyant sur STOP



Zone de paramétrage de la quantification

Sélection de l'affichage temporel et de la partie sonore écoutée : signal quantifié ou bruit de quantification



Représentation temporelle et spectrale
dans le panneau de droite, en fonction
de la sélection faite par l'utilisateur

Quantization Parameters

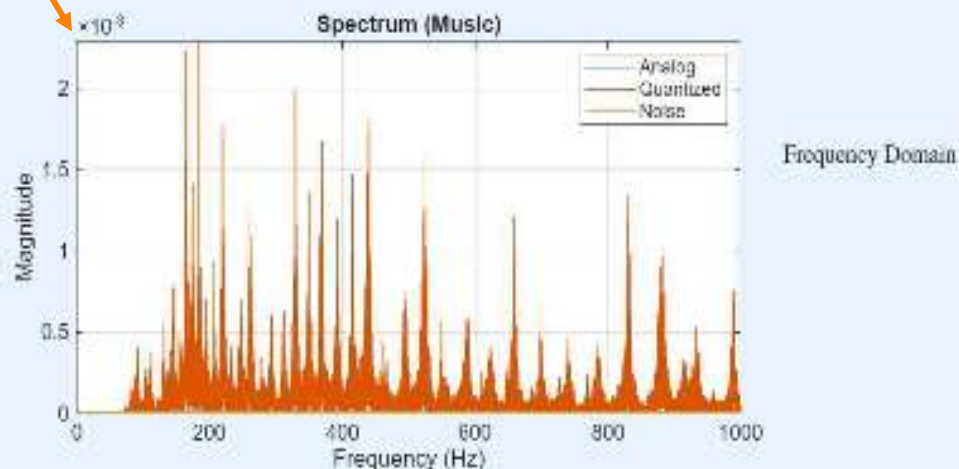
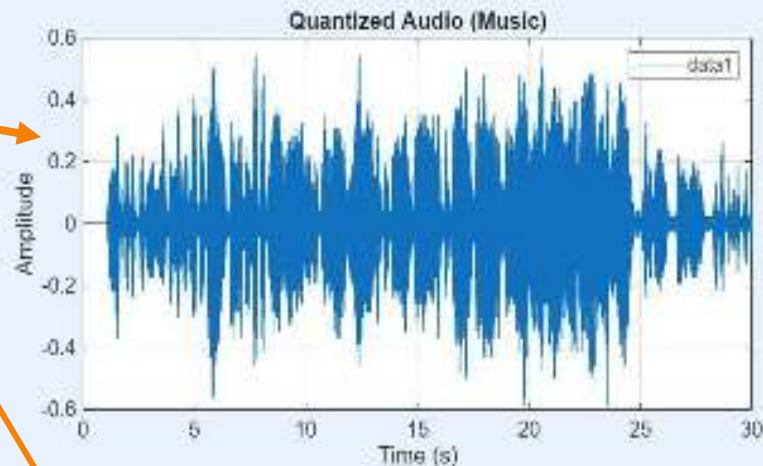
Bit Depth: Mid-rate ☐ Mid-band ☐
Select Uniform Quantization Type

Measurements

Measured Quantization SNR (dB): Theoretical SNR (dB):

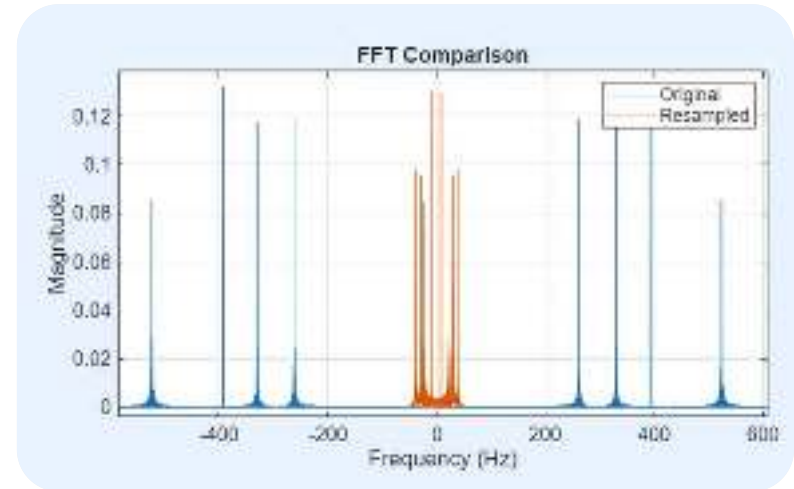
Spectral Analysis

☒ Analog Music
☒ Quantized Music
☒ Quantization Noise



Sous échantillonnage et repliement spectral

- Le critère de Shannon recommande que $f_e > 2f_{\max}$ afin d'éviter les effets de repliement spectral.
- Le sous-échantillonnage consiste à diminuer artificiellement la fréquence d'échantillonnage. Si la nouvelle fréquence est inférieure à $2 f_{\max}$ alors certaines fréquences situées au-delà sont repliées dans la bande plus basse.



Filtrage numérique

- Filtrage numérique = modification du contenu fréquentiel d'un signal afin d'atténuer ou de conserver certaines bandes de fréquences.
- Filtre : de Butterworth, caractérisé par une réponse en fréquence monotone dans la bande passante et une atténuation progressive au-delà de f_c .

Fonction de transfert du filtre dans le domaine en z , caractérise complètement le comportement du filtre


$$H(z) = \frac{Y(z)}{X(z)}$$

$$H(z) = K \frac{\prod_{i=1}^M (z - z_i)}{\prod_{k=1}^N (z - p_k)}$$

Fréquence de coupure normalisée d'un filtre numérique, exprimée par rapport à la fréquence de Nyquist afin de la ramener sur une échelle comprise entre 0 et 1.


$$W_n = \frac{f_c}{f_s/2}$$

Open Existing Noise ...

Single frequency Music Sampling Resampling

Signal Parameters

Import

Sampling Frequency: 44100

Duration (sec): 00:00:00

PLAY

Resampling

Edit Resampling Frequency (Hz): 1000

Apply

STOP

PLAY

PLOT FFT

Filtering

Cut Off Frequency (Hz): 1000

Second roll off freq (Hz): 0

Select Filter Type: Passes band

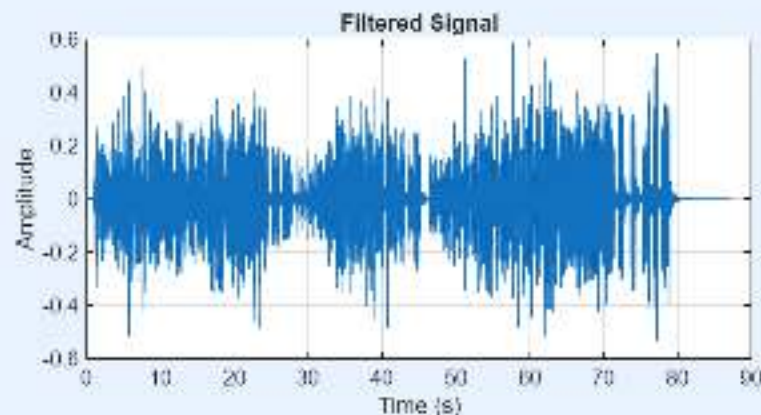
Filter Order: 2

Apply Filter

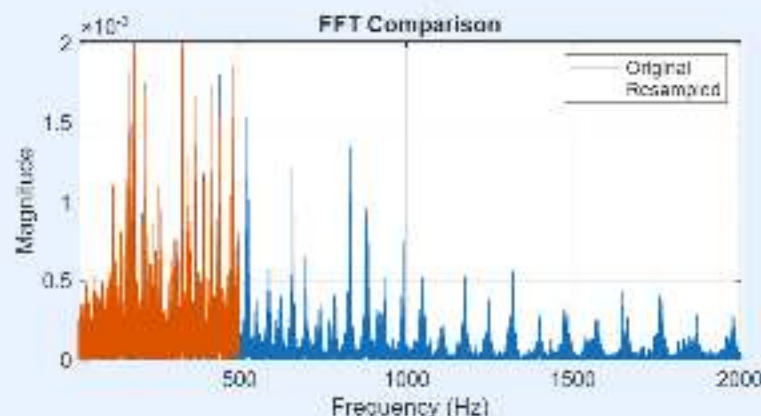
Play Filter

Troisième onglet : Ré-échantillonnage et filtrage

Time Domain



Frequency Domain



Reconstruction du signal

- La reconstruction d'un signal consiste à retrouver une approximation continue à partir des échantillons discrets $x[n] = x(nT_e)$
- Si la condition de Shannon est respectée, le signal peut être reconstruit parfaitement à partir de ses échantillons.

- La formule de l'interpolation de Shannon est :
$$x_r(t) = \sum_{n=-\infty}^{+\infty} x[n] \operatorname{sinc}\left(\frac{t - nT_e}{T_e}\right)$$

où $T_e = \frac{1}{f_s}$ est la période d'échantillonnage et $\operatorname{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$.

Perspectives

- Offrir plus de fonctionnalités à l'utilisateur
- Permettre la reconstruction et le traitement temps réel de signal de parole
- Améliorer l'ergonomie et l'interface utilisateur



Version possible de l'amélioration de l'ergonomie et de l'interface
(image réalisée par IA)

Merci de votre attention