



UE MU4RBI08-audio

Compte-rendu de TP 4 : Compression d'un Signal Audio

NEHMAR Anis, TOUZARI Lisa
Master Automatique, Robotique, Parcours: ISI

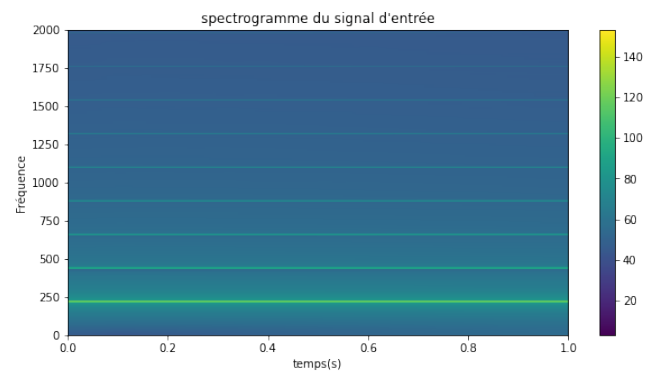
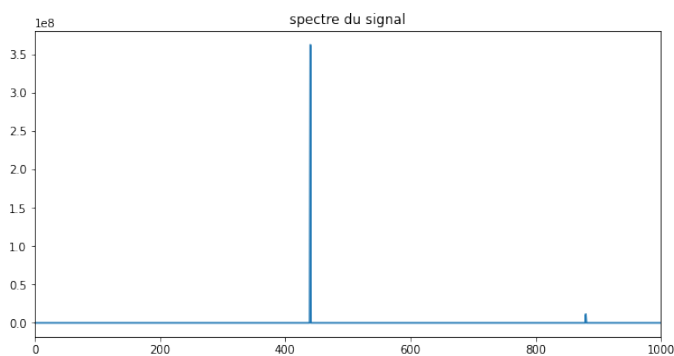
Introduction

La compression audio est une forme de compression de données qui a pour but de réduire la taille d'un flux audio numérique en vue d'une transmission. Le but de ce TP, est donc d'implémenter une méthode de compression audio avec perte qui est inspirée de la norme MPEG-1 layer III.

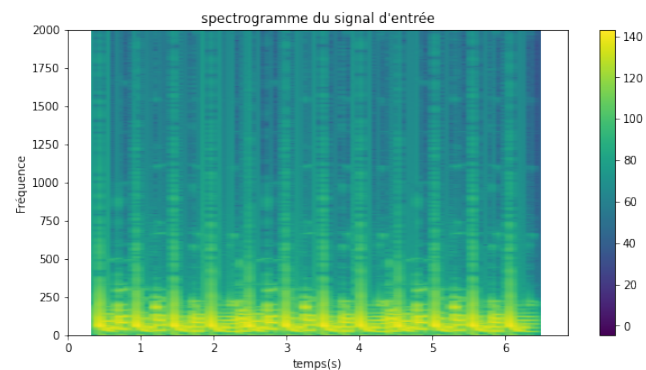
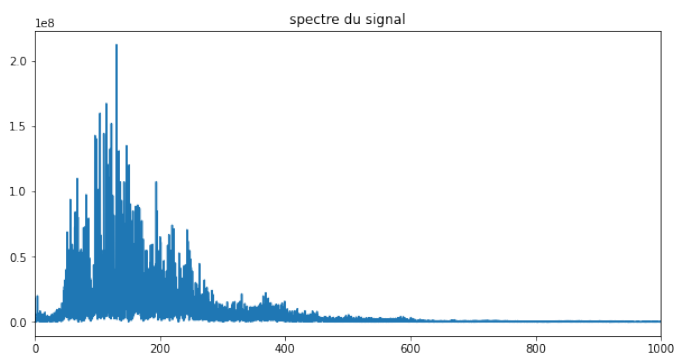
Partie 1 : Codage audio perceptif

1. Calculer la TFCT X du signal audio et tracer le spectrogramme correspondant

(a) Pour le signal **sine440.wav**



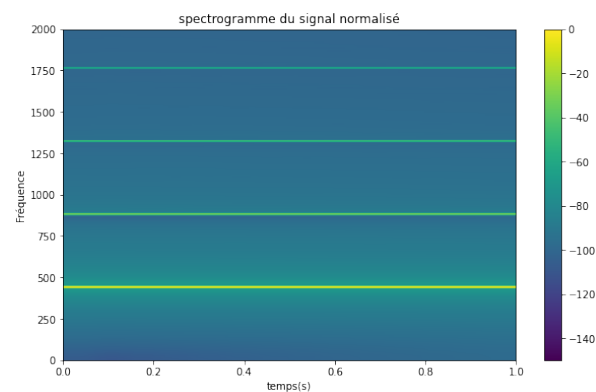
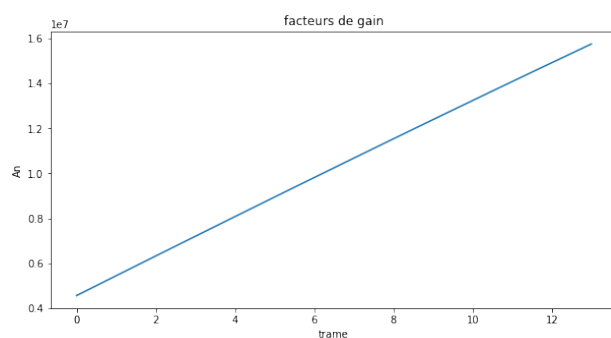
(b) Pour le signal **daftPunkaroundTheWorld.wav**



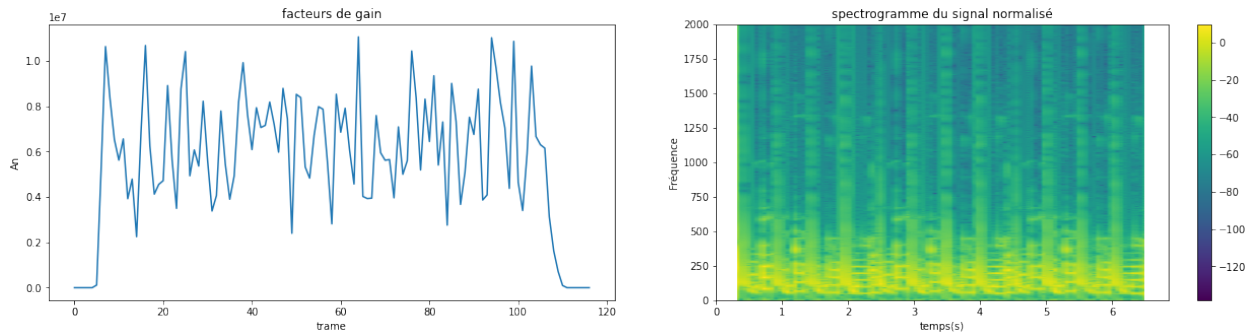
2. calculer le facteur de gain A_n

On normalise notre signal de sorte à avoir la valeur maximale à 1. C'est à dire, on divise chaque trame du spectrogramme par sa valeur maximale.

(a) Pour le signal **sine440.wav**



(b) Pour le signal **daftPunkaroundTheWorld.wav**



3. Calculer, à partir d'un débit souhaité de 392 kbits/s, le nombre de bits alloué à chaque trame du module de la TFCT.

Etant donné un débit de 392 kbits/s, il nous faudra alors déterminer le nombre de bits par trame, pour cela il suffit de calculer le nombre maximale des trames qui nous permet d'atteindre 1 seconde, k en utilisant la relation : $T_e \cdot (N_{win} + k \cdot N_{hop}) = 1 + N_{win} \cdot T_e$. Le nombre de trames par seconde sera alors $k+1$. Donc le nombre de bits par trame sera égal au débit / ($k+1$).

4. Calcul de la répartition des bits en fréquence pour chaque trame, en utilisant l'algorithme d'allocation de bits perceptif

Pour déduire le nombre de bits à allouer pour chaque points fréquentiels des trames du signal on utilise l'algorithme d'allocation de bits perceptif. Dans notre cas :

- Le rapport signal-masque SMR est calculé avec un niveau de masque de 96 db => $SMR = 20 \cdot \log(\text{signal}) - (-96)$.
- Le rapport signal-bruit $SNR = 20 \cdot \log(\text{signal}/\text{bruit})$ tel que : le bruit = signal original - signal quantifié, or comme on n'a pas encore quantifié notre signal on aura un $SNR=0$.
- Le rapport bruit-masque $NMR = SMR - SNR = SMR$.
- On alloue un ensemble de bits deteminé pour chaque trame. Tant qu'on a encore des bits à allouer et que les valeurs du NMR ne sont pas toutes négatives, on determine le point fréquentiel de plus grand module et on lui alloue un bit(avec le nombre de bits maximal pour chaque point est de 16).
- L'allocation d'un bit entrainera une diminution du NMR de 6 db.

5. Expliciter les conditions d'arrêts à utiliser pour l'arrêt de l'allocation

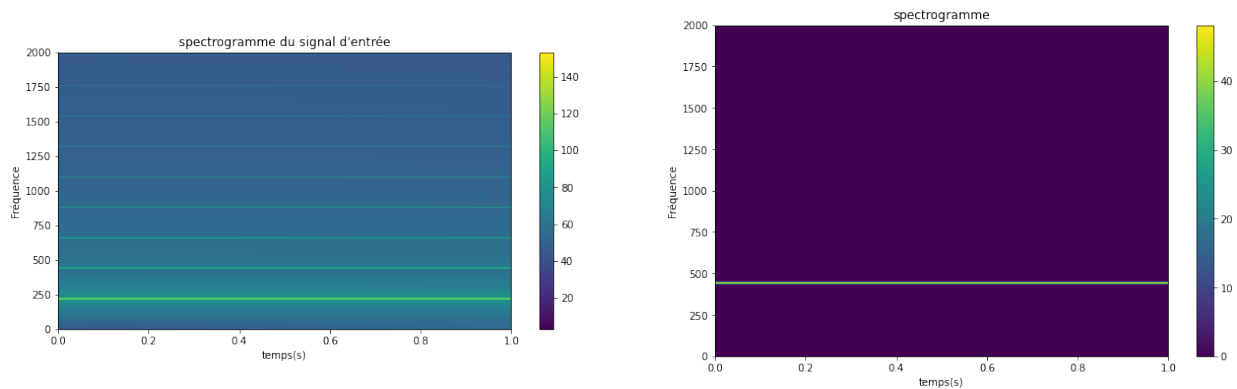
Basé sur la psychoacoustique, tant que le bruit de quantification est inférieur au seuil de masquage à n'importe quelle fréquence, le bruit sera inaudible. D'où la condition d'arrêt sur le NMR (ratio bruit-masque), c'est à dire, dès que toutes les valeurs du NMR sont négatives, on arrête l'allocation des bits.

Une autre condition d'arrêt, c'est qu'on fait une allocation des bits tant qu'on a encore des bits à allouer, d'où la condition $R > 0$.

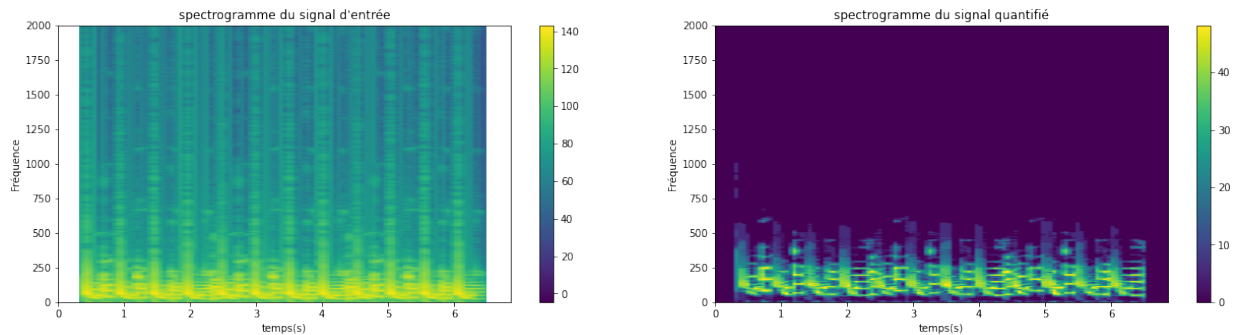
6. déduction du spectrogramme quantifié à partir du nombre de bits donné par les éléments de la matrice Q

On utilise la finction Fuquant pour quantifier notre signal.

(a) Pour le signal **sine440.wav**



(b) Pour le signal **daftPunkaroundTheWorld.wav**



On peut voir d'après le spectrogramme du signal **daftPunkaroundTheWorld.wav** qu'on a pu reconstituer la totalité des fréquences dominantes du signal (information utile) comprise entre 0 et 500 hz. Pour les fréquences supérieures à 500 hz, elles ont été perdus. Lors de l'allocation des bits avec l'algorithme d'allocation perceptif, le priorité est donnée aux fréquences de plus grand module c'est à dire les fréquences dominantes. On attribut plus de bits aux fréquences dominantes qu'aux autres fréquences, on peut voir dans le spectre du signal que les fréquences supérieures à 500 hz sont quasiment nulles, cela implique que le nombre de bits utilisés pour les quantifier sera très petit.

7. restitution de la valeur de chaque point fréquentiel des trames

En appliquant la fonction de déquantification **Fuquant-inv** sur le signal audio quantifié, on pourra recuperer le spectre d'origine normalisé (decompression), pour reconstruire le spectre il faudra alors le dénormaliser en multipliant les différentes trames du spectre par les facteurs de gain A_n calculés précédemment.

8. reconstruction du signal audio compressé y à partir de sa TFCT en utilisant la fonction **tfctinv**

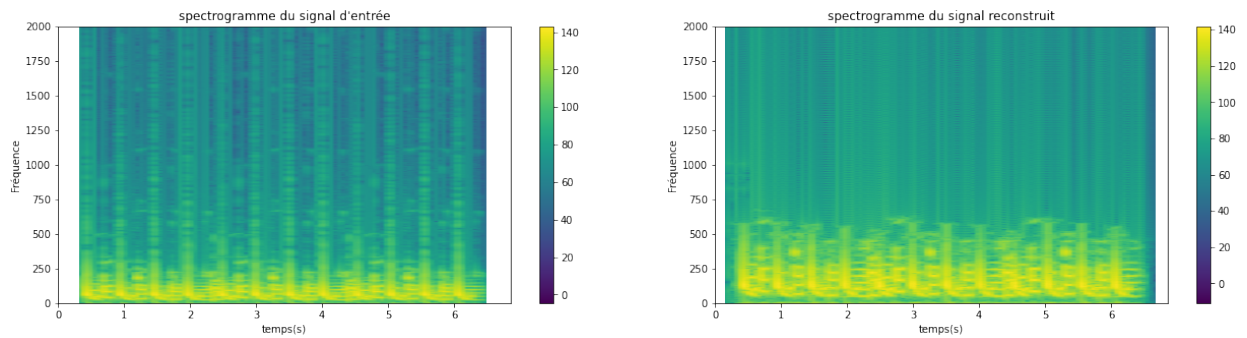
Pour reconstruire le signal audio compressé à partir de sa tfct, on applique directement la fonction tfct inverse sur le spectre après déquantification et dénormalisation.

9. Comparaison de signaux audio restitués après compression et décompression avec différents débits

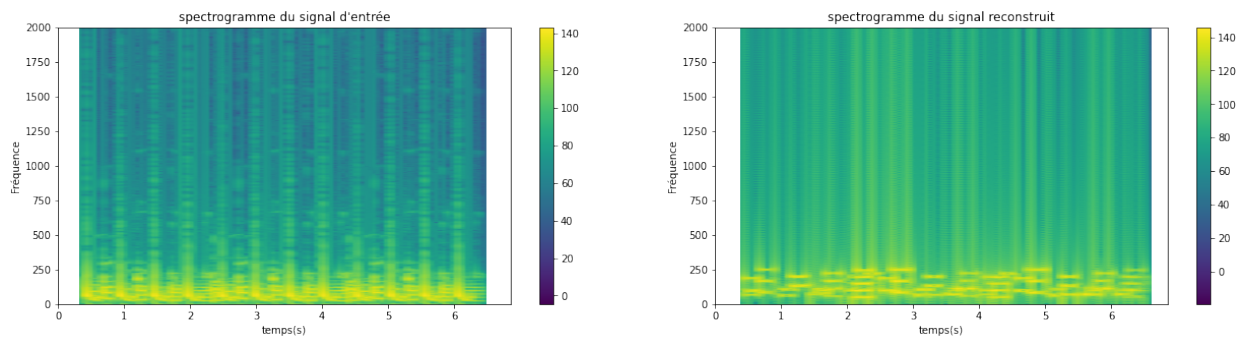
On choisi de restituer notre signal avec un débit de : (1, 10, 100, 392) k.bits/s.
A partir des signaux audio reconstitués, on peut clairement distinguer que la qualité du signal est meilleur plus le débit est grand. Cela est dû au fait que le nombre de bits alloués pour la quantification augmente, donc on pourra quantifier plus de points fréquentiels et on garde plus d'informations utiles.

On compare entre le spectrogramme du signal **daftPunkaroundTheWorld.wav** et du signal reconstruit.

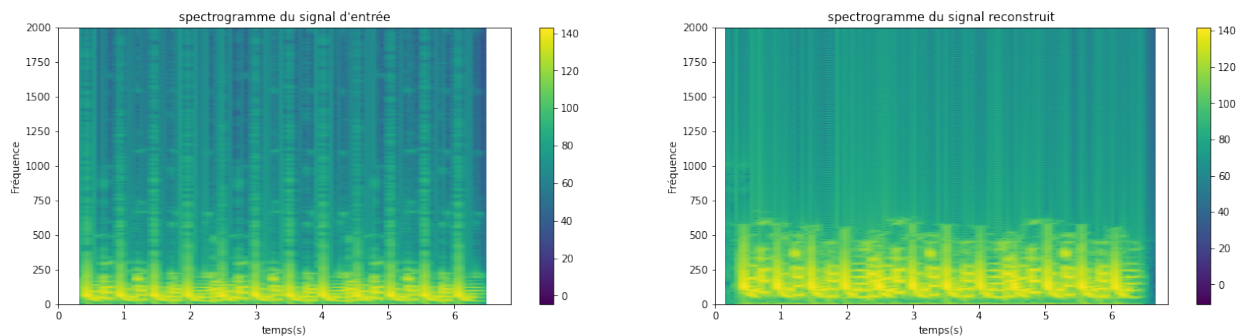
(a) Pour un débit = 392 k.bits/s



(b) Pour un débit = 1 k.bits/s



(c) Pour un débit = 10 k.bits/s



10. Le débit à partir duquel la compression devient transparente

On peut voir que plus le débit augmente, la différence entre le signal reconstitué après compression et le signal d'origine devient plus négligeable.

En augmentant le débit, à partir d'un certain seuil, la compression devient alors inaudible, on ne fait plus de différence entre le signal non compressé et le compressé.

Dans notre cas, pour le signal **daftPunkaroundTheWorld.wav** à partir d'un débit de 100 k.bits/s, le signal reconstitué ne change plus.

11. L'intérêt de l'allocation perceptive de bits par rapport à une allocation uniforme

Pour l'allocation uniforme, on alloue des bits pour tout les points fréquentiels, c'est à dire, dans le cas où en un point le module est nul ou bien très faible (bruit, silence), les bits seront en quelque sorte perdu (pas d'information utile), donc une perte de mémoire de stockage. Pour l'allocation perceptive, on alloue les bits de façon à privilégier les points fréquentiels de plus grande amplitude, c'est à dire là où on a plus d'information, donc on optimise la mémoire de stockage.