TP Bancs de filtres

Roland Badeau

1 Introduction

Ce TP vise à mettre en pratique le filtrage dans un système multicadence (application à la conversion de fréquence d'échantillonnage) et à comprendre et implémenter un banc de filtres à reconstruction parfaite pour l'égalisation audio. Les fichiers nécessaires à ce TP sont disponibles sur le Moodle ATIAM.

2 Conversion de fréquence d'échantillonnage

On désire effectuer la conversion de fréquence d'échantillonnage de $Fe = 48kHz$ vers $Fe = 32kHz$.

1. Décrivez et dessinez la chaîne de traitement numérique qui permettra d'effectuer une telle conversion (on précisera en particulier les caractéristiques du filtre idéal $H(e^{2i\pi\nu})$ qu'il faudrait utiliser).
2. Synthétisez une réponse impulsionnelle $h(n)$ satisfaisante pour cette conversion de fréquence d'échantillonnage. Vous pourrez utiliser les implémentations suivantes de la méthode de Remez :
   — fonction Matlab `firpm` : `[h,delta] = firpm(odd order,2*[0,nu_c,nu_a,.5],[L L 0 0])`
   — fonction Python `remez` : `h = scipy.signal.remez(even length,[0,nu_c,nu_a,.5],[L 0])`
   Tracez la réponse en fréquence $|H(e^{2i\pi\nu})|$ pour vous assurer que votre synthèse est correcte (nous attendons une différence d'au moins 50 dB entre la bande passante et la bande atténuée).
3. Programmez la chaîne de traitement numérique de la question 1. Vous pouvez utiliser la fonction Matlab `filter` ou la fonction Python `scipy.signal.lfilter`.

L'objectif de la suite de cette section est d'obtenir une réalisation optimale de cette conversion de fréquence et de comparer les performances par rapport à une implementation directe.

L'implémentation optimale sera réalisée à l'aide des décompositions polyphases (type I et type II). L'utilisation des identités nobles et de l'équivalence ci-dessous (figure 1) permettra en effet de placer l'opération de filtrage sur des signaux à la plus basse fréquence d'échantillonnage.

4. Vérifier l'équivalence des deux schémas de la figure 1.

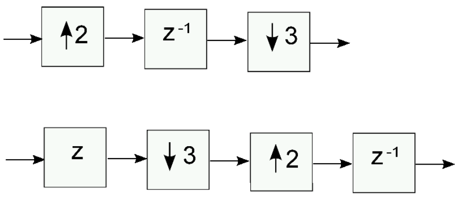

FIGURE 1 – Equivalence

5. Recherchez et programmez l'implémentation efficace de cette conversion de fréquence d'échantillonnage en utilisant deux décompositions polyphases successives ainsi que l'équivalence de la figure 1.
6. Comparez les résultats obtenus par les deux implémentations (directe et efficace) et comparer les temps d'exécution des deux approches (on pourra utiliser les fonctions Matlab `tic` et `toc`, ou la fonction Python `time.time()`).

3 Egalisation audio par TFCT

3.1 Analyse par TFCT

En Matlab, on utilisera le programme `tfct.m` qui fournit un cadre pour le calcul de la transformée de Fourier à Court Terme (TFCT). En Python, on utilisera le modèle de notebook fourni.

La définition de la TFCT dite "en convention passe-bas" est donnée en temps discret par :

$$W_x(\lambda, b) = \sum_{n\in\mathbb{Z}} x(n)w(n-b)e^{-j2\pi\lambda n}, \tag{1}$$

où $w(n)$ est la fenêtre d'analyse en temps discret, supposée sommable, réelle et paire.

1. On choisit pour $w(n)$ une fenêtre de Hann (Hanning) de longueur N_w. Tracer la TFD de w, soit $\hat{w}(k/M)$ où $M \geq N_w$ est l'ordre de la TFD. Que vaut la largeur du lobe principal en fonction de N_w ?
2. Remarquer que l'expression (1), prise à λ fixé, peut s'écrire comme une convolution et en déduire une interprétation de la TFCT en terme de filtrage. Expliciter le rôle du filtre correspondant (passe-bas ? passe-bande ? passe-haut ?). En tant que filtre RIF à phase linéaire, préciser son type (type 1,2,3,4 ?) en fonction de sa longueur (paire ou impaire).
3. Une autre définition de la TFCT, dite "en convention passe-bande" est donnée par

$$\tilde{X}(\lambda, b) = \sum_{n\in\mathbb{Z}} x(n+b)w(n)e^{-j2\pi\lambda n}. \tag{2}$$

   Expliquer cette dernière appellation et calculer $\tilde{X}$ en fonction de W_x. Laquelle des deux conventions correspond au calcul mené dans `tfct.m` ou dans le notebook ?

Dans toute la suite, le calcul de la TFCT sera effectué par TF rapide (FFT) en considérant uniquement des fenêtres causales de longueur finie inférieure à l'ordre M de la TFD. Pour simplifier la TFCT sera maintenant indexée par le numéro k du canal fréquentiel situé autour de $\lambda = k/M$ et l'index temporel u de la trame considérée, soit

$$\tilde{X}(k, u) = \sum_{n\in\mathbb{Z}} x(n+uR)w(n)e^{-j2\pi\frac{kn}{M}}. \tag{3}$$

où R représente le décalage temporel entre chaque trame d'analyse.

4. A l'aide du script Matlab `tfct.m` ou du notebook Python, pour $M = 32$ et $R = 1$, calculer le signal $x_k(u) = \tilde{X}(k, u)$, pour $k = 3$. Est-il réel ou complexe ? Vérifier l'interprétation en terme de filtrage en utilisant les fonctions Matlab `filter` et `spectrogram`, ou les fonctions Python `scipy.signal.lfilter` et `scipy.signal.spectrogram`. Écouter $\mathrm{Re}(x_k)$.

3.2 Reconstruction

La reconstruction s'effectue par une opération *d'addition-recouvrement,* qui s'écrira sous la forme :

$$y(n) = \sum_{u\in\mathbb{Z}} y_s(u, n-uR),$$

où $y_s(u,n) = \mathrm{TFD}^{-1}[\tilde{X}(k,u)](n)\, w_s(n) = \dfrac{1}{M}\sum_{k=0}^{M-1} \tilde{X}(k,u)e^{j2\pi\frac{kn}{M}} \times w_s(n)$

5. Démontrer qu'une condition suffisante de reconstruction parfaite est $f(n) = 1\ \forall n$ où $f(n) = \sum_{u\in\mathbb{Z}} w(n-uR)w_s(n-uR)$. Utiliser le programme `ola.m` ou la fonction `ola` du notebook pour vérifier cette condition pour la fenêtre produit $w_p(n) = w(n)w_s(n) = h(n)^2$, où $h(n)$ est une fenêtre de Hann correctement normalisée, et pour un recouvrement de 75%.
6. Intégrer une partie de resynthèse effectuant l'addition-recouvrement décrite au (e) dans le programme `tfct.m` ou dans le notebook et qui vérifie la reconstruction parfaite.

3.3 Egaliseur Audio à TFCT

On se sert des résultats précédents pour utiliser la TFCT comme un égaliseur à $M/2+1$ canaux réels (on suppose l'ordre M pair). On obtiendra alors les trames de synthèse comme $y_s(u,n) = \mathrm{TFD}^{-1}[\tilde{Y}(k,u)](n)\, w_s(n)$, où $\tilde{Y}$ est obtenu en affectant des poids à chacun des canaux soit $\tilde{Y}(k,u) = w_k\tilde{X}(k,u)$. La condition de reconstruction assure alors $y(n) = x(n)$ lorsque $w_k = 1\ \forall k$.

7. Réaliser l'égaliseur et le tester.