Détection de fréquences fondamentales multiples

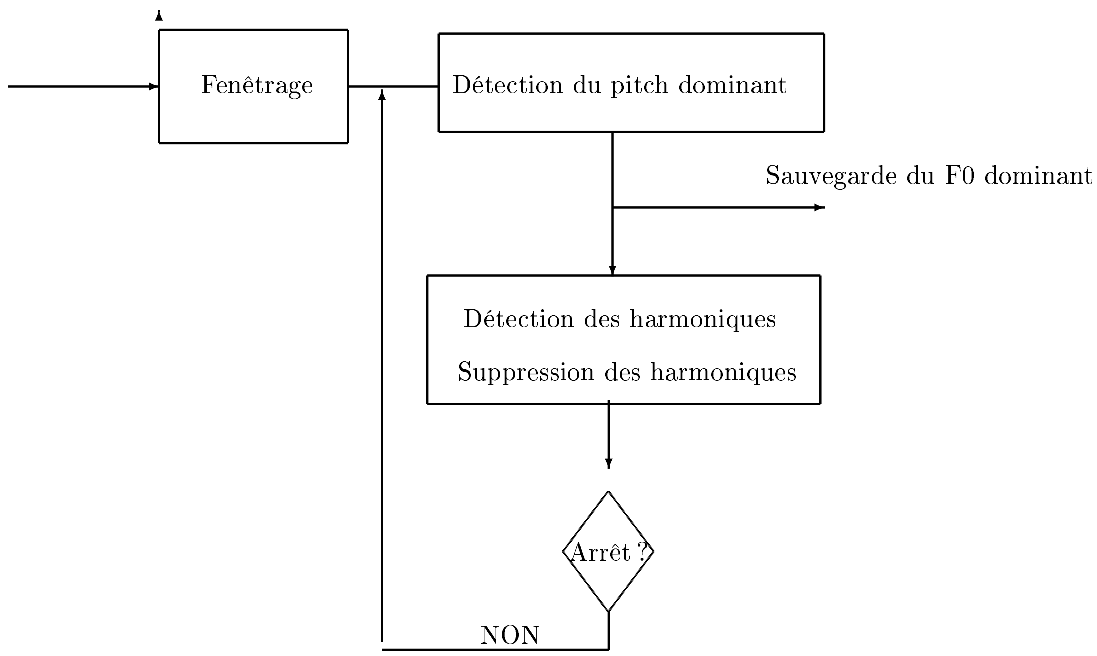
Gaël Richard, Roland Badeau
Fichiers téléchargeables à : `http://www.enst.fr/~grenier/mva/`

L'objectif de ce TP est de réaliser un module d'estimation de fréquences fondamentales multiples d'un signal de musique (ou encore de déterminer à partir du signal audio les notes de musique jouées). S'il existe plusieurs approches possibles pour ce problème, il est proposé ici d'étudier une technique simple inspirée par les travaux de A. Klapuri [1, 2].

Cette approche consiste à estimer dans un premier temps la fréquence fondamentale dominante, à en déduire la position et l'amplitude de ses harmoniques sur le spectre puis à soustraire la contribution du son correspondant. Ce principe est ensuite itéré afin d'extraire l'ensemble des fréquences fondamentales du son initial. Il est ensuite proposé de mettre en place le principe de la continuité de l'enveloppe spectrale (*spectral smoothness*) qui sera appliqué au préalable de la soustraction fréquentielle. Les performances de ces approches seront finalement testées et comparées sur différents accords composés de notes de piano, flute et hautbois.

Ce TP sera réalisé à l'aide du logiciel `MATLAB`. Les sons pourront être chargés avec la fonction `audioread` et écoutés avec la fonction `soundsc`. On trouvera par ailleurs un template pour le programme sur le site mentionné ci-dessus (`TPmultipitch4_template.m`).

La figure résume les différentes étapes de l'algorithme.



1 Fenêtrage et Calcul de Transformée de Fourier rapide

Etudier la première partie du template (`TPmultipitch4_template.m`).

1. Quelle est la taille de la fenêtre utilisée ? vous parait-elle appropriée ?
2. A quoi sert le paramètre offset ?
3. Quelle est la précision fréquentielle en Hertz de la représentation spectrale obtenue après Transformée de Fourier rapide (FFT) ?

2 Estimation de la fréquence fondamentale par la méthode du produit spectral

Nous allons estimer ici la fréquence fondamentale F_0 d'un signal x de longueur N, échantillonné à la fréquence F_s. Cette valeur sera recherchée dans un intervalle $[F_{\min}, F_{\max}]$, avec une précision au moins égale à $\mathrm{d}F$. Nous allons utiliser la méthode du produit spectral, qui sera calculé en multipliant H versions compressées du spectre. Le produit spectral est ainsi donné par :

$$P(e^{2j\pi f}) = \prod_{n=1}^{n=H} |X(e^{2j\pi n f})| \tag{1}$$

Les paramètres de notre estimateur seront :
- dF_{min}, la résolution fréquentielle minimale (par défaut $dF_{min} = Fs/N$ où N est la taille de la fenêtre d'analyse) ;
- F_{min} et F_{max} donnant l'interval de recherche des fréquences fondamentales (on pourra prendre $F_{\min} =$ 50 Hz, $F_{\max} = 900$ Hz)
- `H`, le nombre de versions compressées utilisées (on pourra prendre $H = 4$)

2.1 Calcul du spectre du signal *(réalisée dans le template)*

Nous allons calculer la transformée de Fourier discrète du signal x sur N_{fft} points. On commencera par multiplier x par une fenêtre de hamming de taille N pour diminuer la hauteur des lobes secondaires. Déterminer ensuite la valeur minimale de N_{fft} pour obtenir une précision au moins égale à $\mathrm{d}F_{min}$ (on choisira en particulier une puissance de 2 pour que l'algorithme rapide puisse être appliqué, que l'on calculera à l'aide de la fonction `nextpow2`). Calculer ensuite la transformée de Fourier Discrête $X(k)$.

2.2 Calcul du produit spectral

Le produit spectral P sera codé en `MATLAB` dans un vecteur de longueur R, couvrant l'intervalle $[0, R-1]$. La fréquence maximale intervenant dans le calcul de P sera donc $H\frac{R-1}{N_{\text{fft}}}Fs$. En déduire, en fonction de N_{fft}, la valeur maximale de R qui garantit de ne pas dépasser la fréquence de Nyquist ($\frac{Fs}{2}$). Calculer ensuite P en fonction de $|X|$.

2.3 Recherche du maximum du produit spectral

Déterminer les valeurs entières $N_{\min}$ et $N_{\max}$ qui correspondent à l'intervalle $[F_{\min}, F_{\max}]$ (on veillera à ce que $N_{\max}$ reste inférieur à R). Rechercher le maximum de P sur l'intervalle $[N_{\min}, N_{\max}]$, et en déduire la valeur de la fréquence fondamentale F_0. On testera la fonction de détection de la fréquence fondamentale sur des signaux monophoniques (par exemple `A4_piano.wav ou E4_oboe.wav).`

3 Soustraction du son correspondant à la fréquence fondamentale détectée

3.1 Détection des harmoniques

Dans un premier temps, il s'agit de détecter les harmoniques du son considéré. La procédure qui est proposée consiste à rechercher le maximum du spectre autour de chaque harmonique théorique de fréquence $f_k = k.F_0$. L'intervalle de recherche autour de chaque harmonique est donné par $[f_{kmin} : f_{kmax}]$. On pourra choisir $f_{kmin} = (1-\alpha) * f_{inharmo}$ et $f_{kmax} = (1+\alpha) * f_{inharmo}$ avec $f_{inharmo} = k * F_0 * \sqrt{(1 + k^2 * \beta)}$ où β est le coefficient d'inharmonicité (on pourra prendre $\beta = 0$ comme valeur par défaut). On pourra par ailleurs appliquer cette recherche également pour $k = 1$ afin d'affiner l'estimation de F_0.

Quel choix de α, β vous parait judicieux pour le piano ? pour le hautbois ? Ce modèle de recherche des harmoniques vous parait-il judicieux ? justifier.

3.2 Suppression des harmoniques

Il s'agit ensuite de supprimer la totalité des harmoniques correspondant à un son donné. Pour cela, il suffit de calculer la largeur théorique d'un pic spectral en fonction de la largeur de la fenêtre d'analyse utilisée. Ensuite, on annule l'ensemble des bins fréquentiels correspondant à chaque harmonique (i.e sur l'intervalle $[k1 : k2]$). En raison de la technique d'estimation de la fréquence fondamentale utilisée, la mise à zéro des harmoniques perturbe les itérations ultérieures de l'algorithme et on remplacera avantageusement la mise à zéro par le forçage des valeurs du spectre autour de chaque harmonique au minimum du spectre sur cet intervalle $(|X_k(k1 : k2)| = \min(|X_k(k1 : k2))|)$.

3.3 Critère d'arrêt

Les étapes précédentes sont ensuite itérées tant qu'un certain critère d'arrêt n'est pas vérifié. Déterminer un critère d'arrêt qui vous permet d'arrêter l'itération lorsque le bon nombre de notes a été trouvé. On testera l'algorithme sur les différents accords proposés.

3.4 Suppression des harmoniques avec application du principe du "spectral smoothness" (*Facultatif*)

Afin de mieux discerner les sons qui sont en relation harmonique (et notamment les sons à l'octave) il est préférable de ne pas soustraire totalement un son de la mixture, mais d'essayer de ne soustraire que sa contribution au niveau de chaque harmonique. Pour cela, on utilise le principe de la continuité de l'enveloppe spectrale (*"spectral smoothness"*) qui consiste à calculer un spectre harmonique lissé où l'amplitude au niveau de chaque harmonique f_k est remplacée par la moyenne des amplitudes des harmoniques f_{k-1}, f_k et f_{k+1}. Ensuite, la soustraction spectrale consiste à :

- forcer les valeurs du spectre autour de chaque harmonique au minimum du spectre sur cet intervalle $(|X_k(k1 : k2)| = \min(|X_k(k1 : k2))|)$ si la valeur du spectre au niveau de l'harmonique est inférieure à celle obtenue sur le spectre harmonique lissé ;
- forcer les valeurs du spectre autour de chaque harmonique à la différence entre le spectre et le spectre lissé sur cet intervalle $[k1 : k2]$ si la valeur du spectre au niveau de l'harmonique est supérieure à celle obtenue sur le spectre harmonique lissé ;

Variante : Une variante de la méthode précédente pourra également être évaluée en calculant un spectre harmonique lissé où l'amplitude au niveau de chaque harmonique est remplacée par la moyenne pondéré des amplitudes des harmoniques entourant l'harmonique considérée f_k. La pondération sera ici effectuée en appliquant une fonction triangle d'une largeur d'un octave centrée sur chaque fréquence f_k. Le principe de soustraction est ensuite le même que précédemment.

Références

[1] A.P. Klapuri. Multipitch estimation and sound separation by the spectral smoothness principle. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, May 2001.

[2] A.P. Klapuri. Multiple fundamental frequency estimation by harmonicity and spectral smoothness. *IEEE Trans. Speech and Audio Processing*, 11(6) :804–816, 2003.