

## Traitement Numérique du Signal BE

Ce BE de traitement numérique du signal (composé de 4 séances de 1h45) traite de l'analyse spectrale de signaux numériques et du filtrage numérique mettant en oeuvre des filtres à réponse impulsionnelle finie (RIF). La notation porte sur un rapport rendu en binôme à l'issue des 4 séances. Ce BE se décompose en trois parties :

- la première partie (2 séances de 1h45) est consacrée à l'analyse spectrale. Il s'agit de compléter le fichier matlab à trous `BE_TNS_Part1_AnalyseSpectrale.m`, disponible dans le répertoire suivant :  
`ServeursFichiers(n7fs)\be\en\TP_TSMatlab\TNS_1EEEE`.
- la deuxième partie (1.5 séance environ) est consacrée à l'étude des filtres RIF (méthode de synthèse et propriétés). L'interface matlab `filtnum.m` permet de réaliser la synthèse de filtres RIF après spécification d'un gabarit, d'observer leurs caractéristiques et de filtrer des signaux synthétiques. L'énoncé associé est détaillé ci-dessous (Partie 2), après quelques rappels utiles sur les filtres RIF.
- la troisième et dernière partie (0.5 séance environ) est consacrée au filtrage du signal réel `ventilateur_Fe_5120Hz.txt`. Il s'agit d'expérimenter, sur ce signal, la fonction matlab `filter` en utilisant les coefficients obtenus à partir de l'interface `filtnum.m`.

### 1 Partie 1 : Analyse spectrale

Compléter le fichier matlab à trous `BE_TNS_Part1_AnalyseSpectrale.m`, disponible dans le répertoire `ServeursFichiers(n7fs)\be\en\TP_TSMatlab\TNS_1EEEE`. L'énoncé (instructions et questions) est intégré à ce fichier matlab sous forme de commentaires. Dans un premier temps, il s'agit de générer un signal puis d'en effectuer l'analyse spectrale à partir de différents estimateurs spectraux. Dans un deuxième temps, les estimateurs spectraux sont appliqués à un signal réel, `ventilateur_Fe_5120Hz.txt`, disponible dans le même répertoire. Il s'agira d'identifier une composante parasite que l'on proposera de supprimer par filtrage dans la dernière partie du BE.

### 2 Partie 2 : Filtre à Réponse Impulsionnelle Finie (RIF)

Cette partie du BE est consacrée à l'étude des filtres à Réponse Impulsionnelle Finie (RIF). Dans le cas des filtres RIF, tout échantillon du signal en sortie est la somme pondérée d'échantillons du signal en entrée. Les filtres RIF sont fréquemment désignés par le terme de filtres non-récurrents, car ils ne présentent pas de boucle de réaction de la sortie vers l'entrée. Ils peuvent être synthétisés directement par un développement en série de Fourier du gabarit idéal.

#### 2.1 Rappels

##### 2.1.1 Définition

Ce sont des systèmes à réponse impulsionnelle finie, de fonction de transfert  $H(z)$ , dont les coefficients  $h(k)$  sont tels que :

$$\begin{aligned} h(k) &\neq 0 \text{ pour } k \in [0, N-1], \\ h(k) &= 0 \text{ sinon,} \end{aligned}$$

$N$  étant l'ordre du filtre c'est-à-dire le nombre de coefficients du filtre non nuls. On obtient l'expression de la fonction de transfert dans le plan des  $z$  :

$$H(z) = \sum_{k=0}^{N-1} h(k) z^{-k}.$$

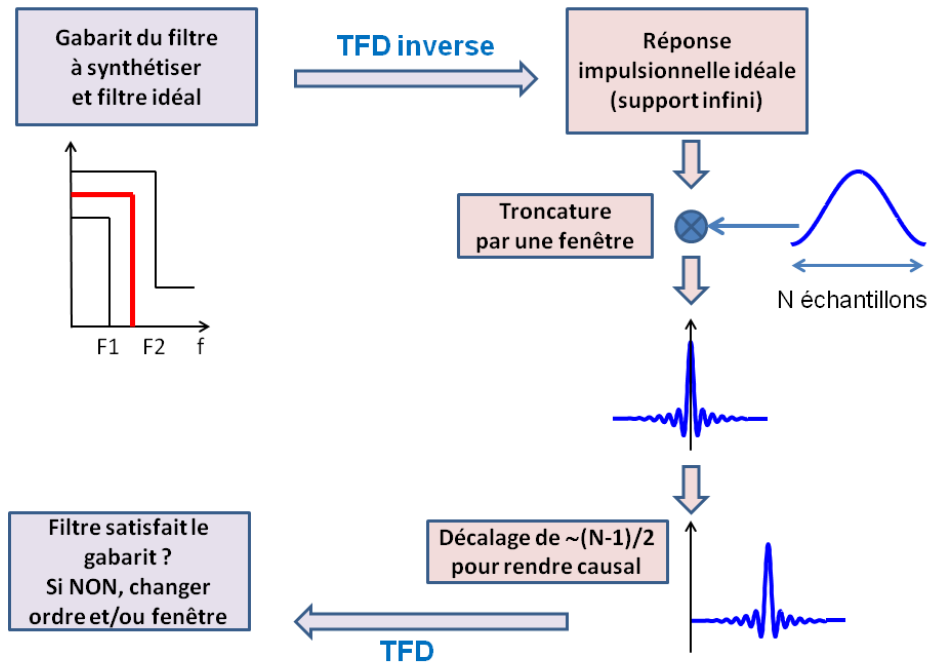


FIGURE 1 – Etapes de la synthèse d’un RIF.

Le caractère non récursif apparaît clairement dans l’expression temporelle reliant la sortie à l’entrée et à la réponse impulsionnelle du filtre :

$$y(n) = \sum_{k=0}^{N-1} h(k) x(n-k).$$

La méthode de synthèse des filtres RIF permet d’assurer une phase linéaire, caractéristique recherchée dans de nombreuses applications. Ceci implique une symétrie de la réponse impulsionnelle. En effet, on veut :

$$H(f) = R(f) e^{j\Phi(f)}$$

avec  $R(f) \in \mathbb{R}$  et  $\Phi(f) = -2\pi f\tau_{pg}$ . La réponse impulsionnelle d’un tel filtre s’écrit :

$$h(t) = \int_{-\infty}^{+\infty} R(f) e^{j\Phi(f)} e^{j2\pi ft} df = \int_{-\infty}^{+\infty} R(f) e^{j2\pi f(t-\tau_{pg})} df.$$

On décompose  $R(f)$  en la somme d’une partie paire  $R_p(f)$  et d’une partie impaire  $R_i(f)$ . La réponse impulsionnelle  $h(t)$  étant réelle (dans le cas d’un signal réel le spectre d’amplitude est pair et la phase est impaire), on a :

$$h(\tau_{pg} + t) = 2 \int_{-\infty}^{+\infty} R_p(f) \cos(2\pi ft) df = h(\tau_{pg} - t).$$

Cette relation fait donc apparaître la symétrie de la réponse impulsionnelle par rapport au point  $t = \tau_{pg}$  de l’axe des temps.

### 2.1.2 Synthèse par la méthode de la fenêtre

La synthèse de filtres RIF est simple, intuitive et facile à mettre en oeuvre, illustrée sur la figure 1. On se donne un gabarit fréquentiel  $H(f)$  à respecter. On calcule la réponse impulsionnelle du filtre recherché par transformée de Fourier inverse de ce gabarit. Il s’agit alors de faire une troncature afin de garder un nombre fini  $N$  d’éléments qui seront les coefficients du filtre, puis d’effectuer un décalage afin de rendre le filtre causal c’est-à-dire physiquement réalisable.

*Etapes successives de la synthèse :*

1. Définir un gabarit fréquentiel en fréquences normalisées.
2. En faire un développement en série de Fourier (ou TFD inverse) :

$$h(k) = \int_{-\frac{1}{2}}^{\frac{1}{2}} H(e^{j2\pi\tilde{f}}) e^{j2\pi\tilde{f}k} d\tilde{f}.$$

3. Multiplication par une fenêtre temporelle  $w(k)$  de longueur  $N$  (ordre du filtre) avec  $w(k) = 0$  pour  $|k| > N$ .
4. Réaliser un décalage (translation) afin de satisfaire la condition de causalité. La valeur du temps de propagation de groupe est :

$$\tau_{pg} = -\frac{1}{2\pi} \frac{d\Phi(\tilde{f})}{d\tilde{f}} = \frac{N-1}{2}.$$

## 2.2 Travail à effectuer

Le logiciel se trouve dans `ServeursFichiers(n7fs)\be\en\TP_TSMatlab\TNS_1EEEE`. Récupérer le logiciel `filtnum.m` et le fichier `ventilateur_Fe_5120Hz.txt` dans votre répertoire de travail. Lancer `filtnum.m` depuis ce répertoire sous Matlab.

### 2.2.1 Introduction

L'interface `filtnum.m` permet de synthétiser plusieurs types de filtres numériques dont les filtres RIF. Dans le cadre de ce BE, on ne s'intéressera qu'à ce type de filtres vus en cours. Les filtres RIF peuvent être calculés par un développement en série de Fourier avec différentes fenêtres (rectangulaire, triangulaire, de Hamming et de Kaiser). L'intérêt dans ce BE est de bien comprendre la méthode de synthèse des filtres RIF, l'influence des paramètres (ordre du filtre, fenêtre de troncature ou de pondération) ainsi que les caractéristiques de ces filtres.

### 2.2.2 Gabarit

Le gabarit du filtre est défini par l'utilisateur parmi les quatre catégories suivantes : (1) filtre passe-bas, (2) filtre passe-haut, (3) filtre coupe-bande et (4) filtre passe-bande. On peut régler  $F_e$  la fréquence d'échantillonnage,  $\Delta f$  la largeur de la bande de transition,  $\delta_1$  l'amplitude des ondulations en bande passante et  $\delta_2$  l'amplitude des oscillations en bande atténuée, ces amplitudes étant exprimées en échelle linéaire, comme illustré sur la figure 2.

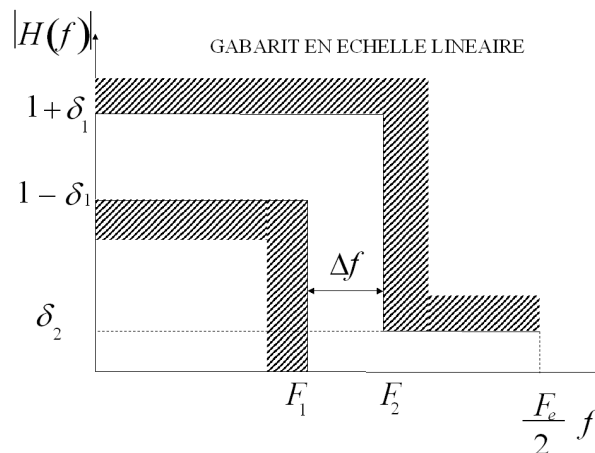


FIGURE 2 – Caractéristiques d'un gabarit fréquentiel de filtre (passe-bas).

En pratique, dans le logiciel utilisé pour le BE, on règle

- la fréquence d'échantillonnage  $F_e$ ,
- les fréquences min et max, définissant la ou les bande(s) de transition notées  $(F_1, F_2)$  pour un passe-bas ou un passe-haut et  $(F_0, F_1) - (F_2, F_3)$  pour un passe-bande ou un coupe-bande (voir figure 3),

n°	Type	$F_e$ (Hz)	$F_0$ (Hz)	$F_1$ (Hz)	$F_2$ (Hz)	$F_3$ (Hz)	$dp$ (dB)	$da$ (dB)
1	Passe-Bas	8000	-	1000	1500	-	3	30
2	Passe-Haut	8000	-	2500	3000	-	3	30
3	Coupe-Bande	8000	1000	1500	2500	3000	3	30
4	Passe-Bande	8000	1000	1500	2500	3000	3	30

TABLE 1 – Exemples de gabarits à tester.

— et les ondulations en bande passante et en bande affaiblie exprimées en dB :

$$dp = 20 \log_{10} \left( \frac{1 + \delta_1}{1 - \delta_1} \right) \text{ et } da = 20 \log_{10} \frac{1}{\delta_2}. \quad (1)$$

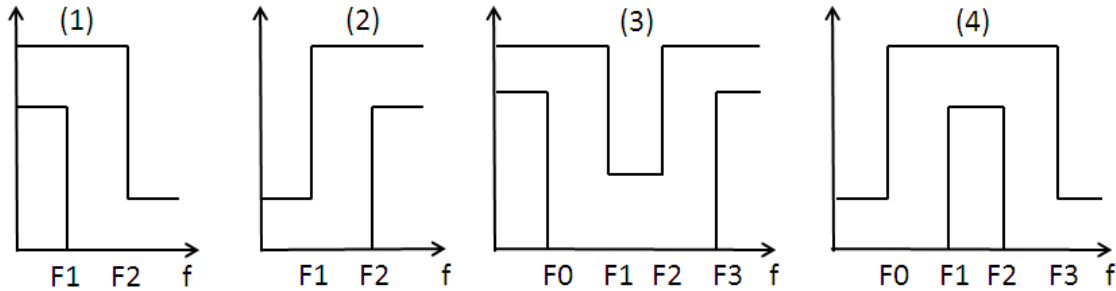


FIGURE 3 – Les 4 types de filtres synthétisés par le logiciel.

A partir d'un gabarit défini en entrant ces différents paramètres, le but de cette partie du BE va être de comparer les résultats de la synthèse des différents filtres pour ce gabarit.

Le gabarit à respecter sera choisi parmi les gabarits du tableau 1 : pour calculer le numéro attribué, faire la somme des mois de naissance des membres du binôme ou trinôme modulo 4 + 1 (sous matlab taper mod(somme,4) + 1).

Pour rentrer les paramètres modifiés, taper dans la fenêtre prévue à cet effet directement la formule en respectant les minuscules et majuscules, sans espace et sans unité : par exemple

$$Fe=8000 \text{ ou } da=30$$

*Remarque préalable : l'utilisateur du logiciel peut choisir l'ordre du filtre à synthétiser. Notons que pour un ordre 2, la synthèse par la fenêtre triangulaire ne donne rien (problème de définition de fenêtre).*

### 2.2.3 Synthèse par la méthode de la fenêtre

#### 1. Evaluation de l'ordre :

Calculer l'ordre du filtre RIF nécessaire pour satisfaire le gabarit n'est pas facile. Il est sûr que plus l'ordre augmente, plus le filtre satisfait le gabarit. Plusieurs formules ont été proposées dans la littérature. Possédant les caractéristiques du gabarit à respecter, il est possible de calculer quel devra être l'ordre  $N$  (très approximatif et en général sous-estimé)<sup>1</sup> :

$$\hat{N} = \frac{2}{3} \log_{10} \left( \frac{1}{10\delta_1\delta_2} \right) \frac{F_e}{\Delta f}$$

où  $F_e$  est la fréquence d'échantillonnage,  $\Delta f$  la largeur de la bande de transition,  $\delta_1$  l'amplitude des ondulations en bande passante et  $\delta_2$  l'amplitude des oscillations en bande atténuée.

Calculer l'ordre donné par l'approximation pour le gabarit choisi. Synthétiser le filtre RIF à cet ordre  $\hat{N}$  avec la fenêtre rectangulaire, puis avec les autres fenêtres disponibles triangulaire, Hamming et Kaiser. Qu'en concluez-vous ?

1. Traitement Numérique du Signal, M. Bellanger, Ed Masson, Collection CNET-ENST.

## 2. Influence de la fenêtre :

Afin de bien comprendre l'influence des différentes fenêtres proposées dans la synthèse de filtres RIFs, on se place à un ordre plus grand que le précédent, par exemple on choisit un ordre  $2\hat{N} + 1$ . Synthétiser le filtre RIF à cet ordre  $2\hat{N} + 1$  en utilisant la méthode de synthèse RIF par fenêtre. Observer les fonctions de transfert obtenues par les quatre types de fenêtres disponibles (rectangulaire, triangulaire, Hamming et Kaiser) : valeur de la pente, position et amplitude du premier lobe d'oscillation. Cette comparaison sera faite uniquement d'un **point de vue qualitatif**, en utilisant le bouton "hold" (et ne pas hésiter à aussi à utiliser le bouton "Ylog"). Quels sont les avantages et les inconvénients de chacune des fenêtres ? Observer le temps de propagation de groupe pour chacun des filtres synthétisés. Que voit-on ?

## 3. Influence de l'ordre :

On appelle ordre optimal  $N_{opt}$  d'un filtre numérique, l'ordre minimal tel que le gabarit fréquentiel soit respecté. Synthétiser le filtre RIF à l'aide de la fenêtre rectangulaire en faisant varier l'ordre du filtre depuis la valeur calculée  $\hat{N}$  ( $\hat{N} < N_{opt}$  !) et en augmentant progressivement jusqu'à dépasser  $N_{opt}$ . Mesurer l'influence de l'ordre sur les paramètres suivants :

- la raideur de la pente,
- la position de la fréquence de coupure à  $-3dB$ ,
- la pseudo-période des lobes d'oscillation

Comparer l'allure des ondulations en bande passante et en bande affaiblie. Quelle remarque peut-on faire ?

### 2.2.4 Réponse impulsionnelle

Observer la réponse impulsionnelle d'un filtre RIF (en filtrant une impulsion). Afin de bien la visualiser,

- faire un zoom,
- Tools - Edit plot,
- clic droit sur la courbe - Marker - "+",

Comment retrouve-t-on les coefficients du filtre à partir de la réponse impulsionnelle ? Justifier son aspect symétrique. Retrouver le temps de propagation.

Lorsque vous avez fini d'observer la réponse impulsionnelle, refaire

- clic droit sur la courbe - Marker - "none" (en bas),
- Tools - Edit plot.

### 2.2.5 Filtrage de signaux synthétiques

Un "bruit blanc" est un signal aléatoire dont la densité spectrale de puissance est constante sur toute la bande spectrale considérée : cela signifie que d'un point de vue spectral, ce signal contient toutes les fréquences avec la même puissance. Vérifier la nature du filtre en lui imposant en entrée un bruit blanc (utiliser le bouton "DSP" qui calcule les DSP de l'entrée et de la sortie du filtre).

Filtrer un signal sinusoidal dont la fréquence est dans la bande passante du filtre. Quelle est la nature du signal en sortie du filtre ?

Filtrer une somme de 2 sinus, composée d'un sinus dans la bande passante du filtre et d'un autre sinus dans la bande atténuée. Qu'observe-t-on sur la sortie ?

## 3 Partie 3 : Filtrage d'un signal réel

Le but de cette dernière partie du BE est de construire un filtre approprié pour un signal réel. On veut construire un filtre qui élimine la fréquence parasite présente dans le signal `ventilateur_Fe_5120Hz.txt` étudié en fin de la première partie du BE. On propose d'utiliser un filtre passe-haut, de bande de transition entre 200 et 300 Hz, une fréquence d'échantillonnage de 5120 Hz et des atténuations de -30dB et 3 dB.

A l'aide du logiciel `filtnum.m`, utilisé dans la partie 2 du BE, construire le filtre qui permet de passer dans le gabarit pour un ordre minimum. A l'aide du bouton sauvegarde, sauvegardez les coefficients du filtre dans un fichier appelé par exemple `coefs.mat`. En ligne de commande Matlab, taper : `load coefs.mat`; (Charge le fichier de coefficients si le fichier s'appelle bien ainsi). Le fichier contient deux vecteurs `a` et `b` avec `a=1` et `b` contenant les coefficients du filtre RIF. Vérifiez le en traçant dans une nouvelle figure le vecteur `b` et en le comparant avec la réponse impulsionnelle du filtre visualisée via l'interface `filtnum.m`

On rappelle l'équation de filtrage qui lie l'entrée  $x(n)$  et la sortie du filtre  $y(n)$

$$y(n) = \sum_{k=0}^{N-1} b_k x(n-k). \quad (2)$$

Cette opération est réalisée par la fonction `filter`. Faire `help filter` pour comprendre comment l'utiliser. Reprenez le code matlab à trous `BE_TNS_Part1_AnalyseSpectrale.m` que vous avez complété lors de la première partie du BE et qui permet de charger et d'analyser le signal `ventilateur_Fe_5120Hz.txt`. Insérez à la suite du programme les lignes de commandes permettant de :

- filtrer le signal avec les coefficients calculés précédemment,
- visualiser l'entrée et la sortie du filtre en temporel,
- analyser spectralement le signal filtré,
- comparer graphiquement la DSP du signal d'entrée et celle du signal de sortie.

Commentaires.