

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mohamed BOUDIAF - M'sila**



**Faculté de Technologie**  
**Département de Génie Electrique**

*MODULE : TP Traitement numérique du signal & TP Modélisation et identification des systèmes*

*ANNEE D'ETUDE : 1<sup>er</sup> Année Master Robotique*

*ENSEIGNANT : Dr. HERIZI Abdelghafour*

**TP 6 : Synthèse des filtres RIF et RII sous Simulink (Matlab)**

**I. Objectif du TP :**

- Synthétiser un filtre numérique sous Matlab.
- Filtrer une séquence audio.

**II. Manipulations sous SPTools : Utilitaire sous Matlab pour la génération de filtre**

**II.1 Synthèse du filtre :**

Sous la fenêtre de commande matlab, taper **sptool**. Il s'agit d'une commande matlab qui permet d'exécuter un utilitaire spécialisé dans le filtrage de signaux.

De la fenêtre de dialogue **Sptool** vous allez pouvoir importer et ou exporter des données depuis/vers des fichiers ou la fenêtre de commande matlab. Il est possible de visualiser immédiatement les effets du filtre, visualiser le gabarit et la réponse du filtre synthétisé.

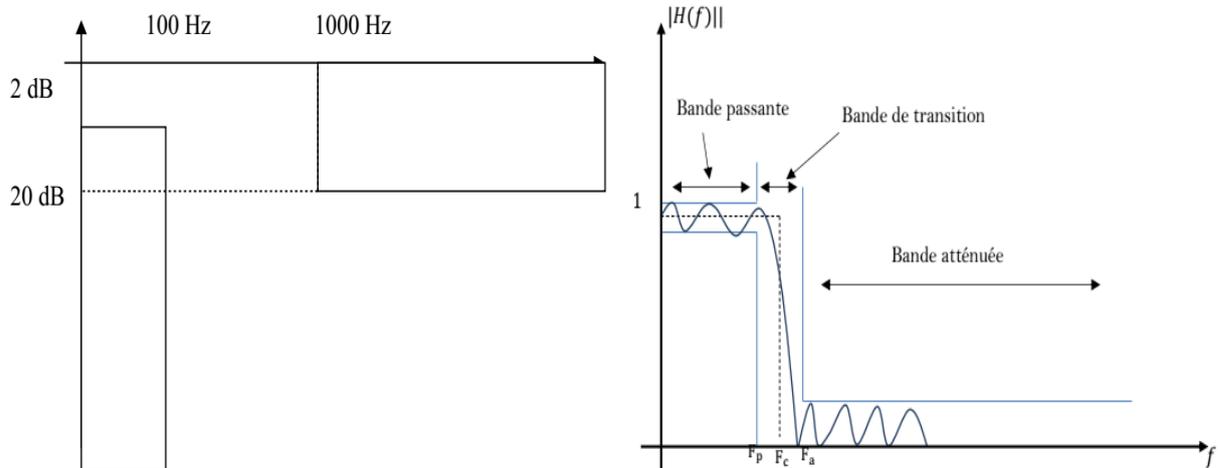
1. Dans la fenêtre SPTool, cliquez sur new design. Une fenêtre Filter Designer s'ouvre.

Saisissez le type de filtre, puis le gabarit ( $F_p$ ,  $R_p$ ,  $F_s$ ,  $R_s$ , Sampling Frequency) et cliquez sur apply.

- $F_p$  bande passante.
- $F_s$  bande coupée.

- Rp atténuation maximale (en dB) dans la bande passante.
- Rs atténuation minimale (en dB) dans la bande coupée.

On souhaite synthétiser un filtre dont le gabarit est le suivant :



Nous choisirons d'abord un filtre « elliptic IIR » avec une fréquence d'échantillonnage de 8192 Hz.

Sélectionnez les bons paramètres, et cliquez sur apply puis sur File/close. Le filtre synthétisé apparaît sous le nom filt1, donnez lui un nom par Edit/Name.

2. Construisez un nouveau filtre de Butterworth de fréquence de coupure 300 Hz et d'ordre 4.

## II.2 Visualisation du filtre :

3. Dans la fenêtre SPTool, choisissez un filtre puis cliquez sur 'View'. Une fenêtre Filter Viewer s'ouvre.
4. Dans la fenêtre Filter Viewer, sélectionnez les réponses que vous voulez observer. A titre d'exemple, nous choisirons la réponse impulsionnelle. Que pouvez vous dire de la réponse du filtre.
5. Trouver les paramètres du filtre.

## II.3 Visualisation des signaux :

6. Synthétiser un signal sin et sincard sur une durée de 2 ms et de fréquence 500 Hz que vous sauvegarderez sous les noms sinus et sinus\_card
7. Récupérer l'extrait 'Toons.wav' et ré-échantillonner le à 8192 Hz à partir du logiciel GoldWave.
8. Dans la fenêtre SPTool, importez ces différents signaux sous File/Import. Choisissez en un ou plusieurs, puis cliquez sur view. Vous devez retrouver l'enveloppe du signal. Vous pouvez aussi jouer la séquence sonore.

## II.4 Filtrage :

9. Avec Filter Designer générer maintenant un filtre passe bande correspondant à la ligne téléphonique (gain de -3dB entre 300Hz – 3400Hz et gain de -40dB à 250Hz et 3450Hz).

10. Indiquez le type de filtre utilisé et son ordre.

Dans la fenêtre SPTool, sélectionnez un signal et un filtre puis cliquez sur apply. Avec le signal Browser, observer le signal avant et après filtrage. Réduisez la bande du filtre (300 – 2000 Hz) et recommencez.

### II.5 Analyse spectrale :

Dans la fenêtre SPTool, sélectionnez un signal, puis dans la colonne Spectre, cliquez sur 'Create'. Dans la fenêtre Spectrum Viewer, choisissez une méthode d'analyse spectrale puis cliquez sur apply.

### III. Manipulations sous Matlab :

Sous matlab, un filtre numérique est représenté par deux vecteurs contenant les coefficients des puissances de  $z^{-1}$ , des polynômes numérateur et dénominateur de sa fonction de transfert en  $z$ .

Un filtre RII de fonction de transfert en  $z$   $H(z)$  donnée par :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$

$H(z)$  peut aussi s'écrire sous la forme :

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}}$$

sera représentée sous Matlab par :

$$Nm = [b_0 \ b_1 \ b_2 \ \dots \ b_M]$$

$$Dn = [1 \ a_1 \ a_2 \ \dots \ a_N]$$

Un filtre RIF de fonction de transfert en  $z$ ,  $H(z)$  donnée par :

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^M b_k z^{-k}$$

sera représentée sous Matlab par :

$$Nm = [b_0 \ b_1 \ b_2 \ \dots \ b_M]$$

$$Dn = [1 \ 0 \ 0 \ \dots \ 0]$$

La commande **impz** permet de calculer la réponse impulsionnelle du filtre.

$$[h, n] = \text{impz}(Nm, Dn)$$

La commande **filter** permet de calculer la réponse temporelle du filtre à une entrée quelconque.

$$yn = \text{filter}(Nm, Dn, xn)$$

La commande **freqz** permet de calculer la réponse fréquentielle du filtre.

$$[H, f] = \text{freqz}(Nm, Dn, N, Fe)$$

Où :  $Fe$  est la fréquence d'échantillonnage et  $N$  est le nombre de points de calcul de la réponse fréquentielle entre  $[0, \frac{Fe}{2}]$ .

### III.1 Manipulation 1 :

Écrire le programme Matlab permettant de calculer et de tracer la réponse impulsionnelle (fonction `impz`), la réponse fréquentielle en module et en phase (fonction `freqz`), le temps de propagation de groupe (fonction `grpdelay`) et les pôles et les zéros dans le plan complexe (fonction `pzmap` et `zplane`), pour chacun des filtres suivants :

- Filtre 1 :  $y(n) = \frac{x(n)+x(n-1)}{2}$
- Filtre 2 :  $y(n) = \frac{x(n)-x(n-1)}{2}$
- Filtre 3 :  $y(n) = 0,1 x(n-1) + 0,2 x(n-2) + 0,1 x(n-3)$
- Filtre 4 :  $y(n) = 0,1x(n) - 0,3x(n-2) + 0,5x(n-3) - 0,3x(n-4) + 0,1x(n-6)$
- Filtre 5 :  $y(n) = x(n) + 0,5 y(n-1)$
- Filtre 6 :  $y(n) = x(n) - 0,5 y(n-1)$
- Filtre 7 :  $y(n) = x(n-1) + 0,5 y(n-1)$
- Filtre 8 :  $y(n) = y(n-1) - 0,25 y(n-2) + x(n) + x(n-1)$

Commenter les résultats obtenus : type de filtre, réponse fréquentielle (passe bas, passe haut, ...), phase du filtre, ...

### III.2 Manipulation 2 :

En utilisant la fonction `filter` de Matlab, calculer et tracer le signal de sortie  $y(n)$  du filtre 3 lorsque l'entrée est :

$$x(n) = \begin{cases} 1 & n = 0,1 \\ 0 & \text{ailleurs} \end{cases}$$

- Calculer théoriquement la sortie du filtre pour  $n = 0 \dots 5$  et vérifier votre calcul avec le résultat donné par la fonction `filter`.

### III.3 Manipulation 3 :

Calculer la sortie du filtre 1 en utilisant la fonction `filter` de Matlab pour un signal sinusoïdal en entrée donné par :

$$x_1 = 1.5 \times \sin(2\pi F_1 \times t) \text{ avec } F_1 = 300 \text{ Hz et } F_e = 10 \text{ kHz}$$

- Tracer les deux signaux d'entrée et de sortie du filtre sur la même figure.
- Commenter.

### III.4 Manipulation 4 :

Toujours en utilisant le même filtre (filtre 1), calculer et tracer le signal de sortie en utilisant la fonction `filter` de Matlab pour le signal d'entrée suivant :

$$x = x_1 + 0.5 \times \sin(2\pi F_2 \times t) \text{ avec } F_2 = 4500 \text{ Hz}$$

- Commenter.

**Annexe :**

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 1.

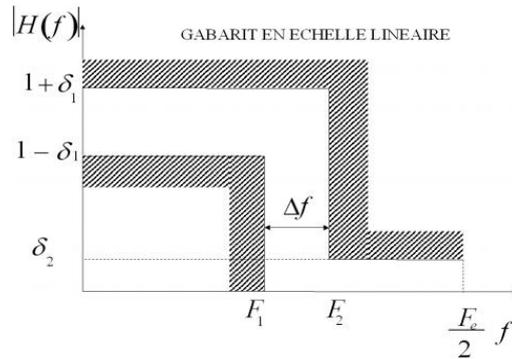


Fig. 1 – Caractéristiques d'un gabarit fréquentiel de filtre (passe-bas).

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

– la fréquence d'échantillonnage  $F_e$ ,

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

Tab. 1 – Exemples de gabarits à tester.

– 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 2) et les ondulations en bande passante et en bande affaiblie exprimées en dB :

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

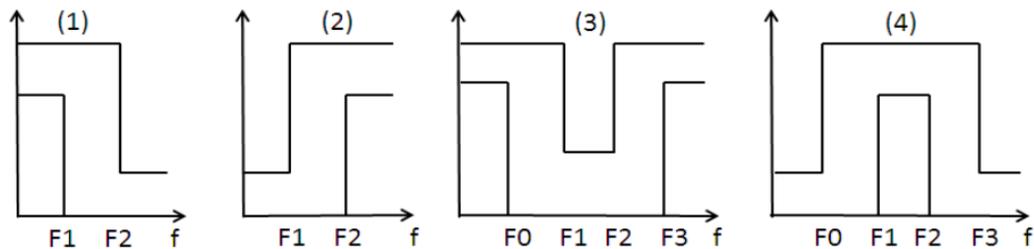


Fig. 2 – Les 4 types de filtres synthétisés par le logiciel.

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

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

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