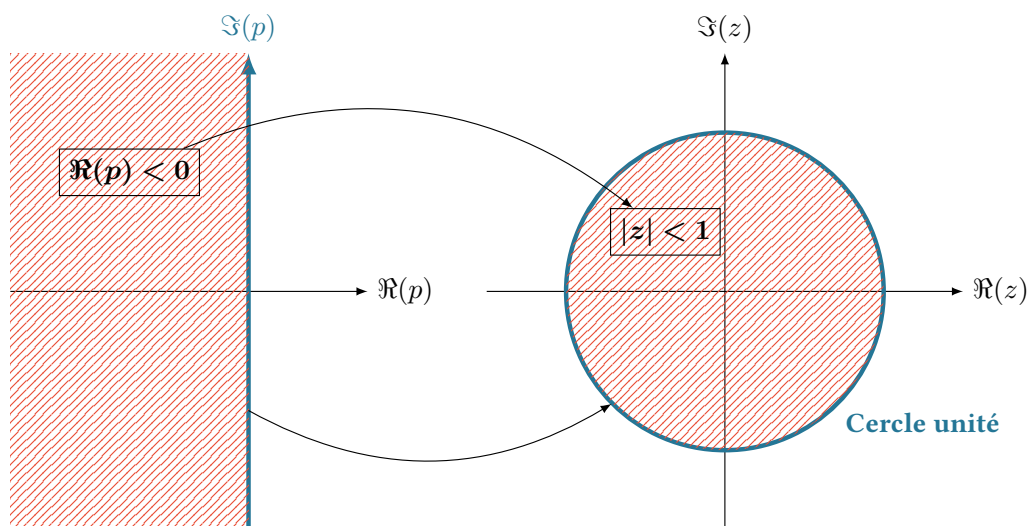


Année universitaire
2020 - 2021

Traitement numérique du signal – 3ETI

TP4

Synthèse de filtres numériques



Eric Van Reeth

1 Présentation du TP

Lors de ce dernier TP de TNS, nous nous intéresserons à la synthèse de filtre dans un contexte concret de débruitage de signal. On disposera pour cela d'un signal de bande maximale 8 kHz échantillonné à la fréquence de 16 kHz et codé sur 16 bits. Ce signal contient une information « utile » fortement perturbée par un bruit large bande. Il va s'agir de concevoir et de caractériser des filtres permettant de révéler le message contenu dans le signal.

1.1 Préparation

Lire l'énoncé et répondre aux questions suivantes :

- Qu'est-ce qu'un spectrogramme ?
- Expliquer le principe de la méthode des fenêtres pour la synthèse de filtre RIF
- Donner une explication pour les paramètres d'entrée et de sortie de la fonction `fir1`
- Expliquer comment on conçoit un filtre RII avec la transformée bilinéaire
- Expliquer le principe du « prewarping », c'est-à-dire de la mise en correspondance des fréquences dans la transformation bilinéaire
- Expliquer les paramètres d'entrée et de sortie des fonctions `ellip` et `bilinear`

1.2 Évaluation du TP

Chaque binôme devra rédiger un compte-rendu. En plus du compte-rendu, **l'intégralité du code développé devra également être rendu.** Le code et le compte-rendu seront compressés dans **une archive (.zip) dont le nom contiendra tous les noms des membres du binôme.** L'archive sera remise sur le dépôt MS Teams dans un **délai d'une semaine après le TP.**

Voici quelques règles à suivre pour la rédaction du compte-rendu :

- Les réponses à la préparation doivent être intégrées dans le compte-rendu
- La notation portera sur votre capacité à répondre à **chacune des questions posées**, mais surtout sur la qualité des **commentaires et justifications qui pourront représenter plus de la moitié des points**
- La présentation, la qualité du français et de l'orthographe seront prises en compte
- Les figures doivent être numérotées, et toutes mesures, justifications et commentaires concernant les figures doivent être rédigés dans le compte rendu en renvoyant précisément à la figure numérotée s'y rapportant
- Le **compte-rendu sera remis au format pdf**
- Le code n'est pas à inclure dans le compte-rendu

2 Analyse du signal et définition du gabarit idéal

Extraire le fichier `signalX.wav` (numéro de signal spécifié par l'intervenant), ainsi que la fonction fournie `sgram.m`. Les placer dans le répertoire de travail. Récupérer les valeurs du signal qui se trouvent dans `signalX.wav` à l'aide de la fonction `audioread`. Écouter le signal avec la fonction `soundsc` (en réglant le volume à un niveau assez faible).

2.1 Analyse du signal initial

Le signal d'intérêt est non-stationnaire au sens où les fréquences qui le composent évoluent au cours du temps. On utilise, pour décrire le comportement de tels signaux, une analyse conjointe en temps et en fréquence appelée spectrogramme. Elle décrit l'évolution au cours du temps de l'énergie du signal en fonction de la fréquence. Le spectrogramme d'un signal $s(t)$ s'exprime par :

$$S(t, \nu) = \left| \int_{-\infty}^{\infty} s(\theta) h(t - \theta) e^{-2i\pi\nu\theta} d\theta \right|^2$$

Ce traitement réalise l'analyse spectrale du signal s vu au travers d'une fenêtre d'observation h glissante. On dispose d'une fonction développée pour ce TP (`sgram`) qui permet de calculer cette grandeur. Elle permet également l'affichage du résultat sous la forme d'une image (voir le début de la fonction `sgram` pour une explication sur son fonctionnement). Calculer le spectrogramme du signal fourni en précisant une dynamique d'affichage de 70 dB.

2.2 Définition du gabarit du filtre idéal

L'objectif est de rejeter le niveau de bruit à au moins 20 dB en dessous du niveau maximum de la densité spectrale du signal, en altérant le moins possible le contenu du message.

1. À partir de l'écoute et du tracé du spectrogramme (à fournir), identifier la bande spectrale correspondant au signal et celle correspondant au bruit
2. Définir en justifiant la nature du filtre à synthétiser pour retrouver le signal ainsi que sa fréquence de coupure en Hz puis en fréquence réduite

3 Synthèse de filtres RIF

On concevra ici des filtres à Réponse Impulsionnelle Finie et à phase linéaire que l'on synthétisera avec la **méthode des fenêtres**.

3.1 Synthèse d'un filtre avec une fenêtre rectangulaire

Utiliser la fonction `fir1` pour synthétiser un premier filtre de longueur M (au choix comprise entre 40 et 50) et en utilisant la fenêtre rectangulaire (fonction `rectwin(M)`).

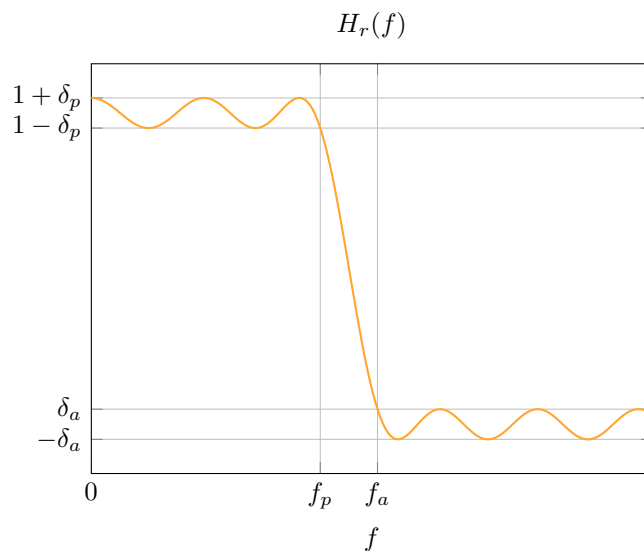
1. Tracer sur une seule figure :
 - la réponse impulsionnelle
 - le diagramme pôle-zéro (fonction `zplane`)
 - le gain complexe en module (tracé en dB) et phase (fonctions `freqz`, `abs`, `angle`, `unwrap`)
2. Vérifier que la fréquence de coupure est bien celle attendue

3.2 Synthèse d'un filtre avec une autre fenêtre

1. Concevoir un second filtre, de même longueur que le précédent, en utilisant une autre fenêtre au choix parmi :
 - Hamming (fonction `hamming(M)`)
 - Hanning (fonction `hann(M)`)
 - Blackman (fonction `blackman(M)`)
2. Effectuer les mêmes tracés que pour le filtre à fenêtre rectangulaire

3.3 Comparaison des caractéristiques des filtres

1. Comparer chacune des caractéristiques pour les deux filtres réalisés
2. Sur les modules du gain complexe tracés cette fois-ci en échelle linéaire, mesurer les fréquences critiques f_a et f_p et les niveaux d'oscillation δ_a et δ_p dans les bandes atténuées et passantes de chacun des deux filtres synthétisés (se référer au schéma ci-dessous pour les mesures)



3.4 Filtrage

1. Filtrer le signal du fichier `signalX` (à l'aide de la fonction `filter`) avec les deux filtres RIF synthétisés
2. Écouter, analyser (`sgram` avec 60 dB de dynamique) et comparer les caractéristiques des signaux de sortie
3. Peut-on répondre au cahier des charges avec ces filtres ?

3.5 Amélioration du filtrage

1. Modifier les paramètres du filtre (nature de la fenêtre, longueur, fréquence de coupure)
2. Que se passe-t-il si on augmente significativement (au-delà de 50) la longueur du filtre ?
3. À partir de quelle longueur et pour quelle fenêtre respecte-t-on le cahier des charges ?
4. Quel est l'effet de la diminution de la fréquence de coupure ?
5. On fournira les spectrogrammes des signaux de sortie pour étayer les commentaires ci-dessus.

4 Synthèse de filtres RII

Toujours pour filtrer le même signal, on va synthétiser un filtre passe-bas à partir d'un filtre analogique passe-bas de type elliptique.

4.1 Conception du filtre analogique

A l'aide de la fonction `ellip`, définir le filtre analogique de départ que l'on transformera en filtre numérique à l'aide de la transformée bilinéaire (`bilinear`).

1. Définir les paramètres W_p , R_p et R_s pour générer la fonction de transfert associée à un filtre analogique elliptique d'ordre 7. On demandera une atténuation dans la bande atténuée d'aussi bonne qualité que celle mesurée avec la seconde fenêtre choisie à la fréquence f_a . Prendre 1 dB d'atténuation à la fréquence de coupure.
2. Calculer le gain complexe de ce filtre (`freqs`) puis afficher son module (en dB) et sa phase.

4.2 Synthèse du filtre numérique

1. Synthétiser le filtre numérique (`bilinear`) en spécifiant que les fréquences de coupure du filtre analogique et celles définies dans le domaine numérique doivent correspondre exactement
2. Tracer et décrire les caractéristiques du filtre numérique obtenu (réponse impulsionnelle obtenue avec `impz`, gain complexe en module et phase, diagramme pôle-zéro, coefficients de la fonction de transfert...)
3. Mesurer les fréquences critiques f_a et f_p et les niveaux d'oscillation δ_a et δ_p dans les bandes passantes et atténuées puis comparer avec les résultats obtenus pour le filtre RIF
4. Filtrer le signal et analyser le résultat (`sgram`, et écoute du signal filtré)

4.3 Optimisation

En fonction du résultat obtenu, on pourra modifier l'ordre (légèrement) ou la fréquence de coupure ainsi que les paramètres d'atténuation, pour proposer un filtre plus efficace.