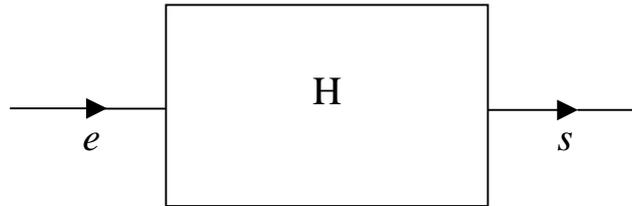


TP de FILTRAGE



TP DE FILTRAGE	1
1 ETUDE DE FILTRES NUMÉRIQUES	2
1.1 GABARIT D'UN FILTRE.....	2
1.2 FONCTIONS MATLAB	2
1.3 APPLICATION.....	2
2 ESTIMATION DE GAIN COMPLEXE	3
2.1 INTRODUCTION.....	3
2.2 OUTILS :	3
2.3 ALGORITHME :	3
2.4 SIMULATION.....	3

1 Etude de filtres numériques

1.1 Gabarit d'un filtre

Le gabarit d'un filtre n'est autre que l'ensemble des caractéristiques du filtre, à savoir :

- ✓ Le gain du filtre dans la bande passante.
- ✓ L'atténuation du filtre en bande coupée.
- ✓ La fréquence de coupure f_c , on l'exprime souvent sous forme normalisée par rapport à la fréquence d'échantillonnage. Elle vérifie $|\text{TF}\{h\}(f_c)| = \max\{|\text{TF}\{h\}(f)|\} / 2^{1/2}$
- ✓ La largeur de bande de transition souhaitée qui doit être la plus petite possible.
- ✓ Les éventuelles oscillations en bande passante et/ou atténuée.

1.2 Fonctions MATLAB

Soit $h(z)$ la transformée en z d'un filtre numérique donné dont la décomposition sous forme fraction rationnelle est donnée par :

$$h(z) = \frac{b(1) + b(2)z^{-1} + \dots + b(nb+1)z^{-nb}}{1 + a(2)z^{-1} + \dots + a(na+1)z^{-na}}$$

- ✓ « $\underline{s} = \text{filter}(\underline{b}, \underline{a}, \underline{e})$ » filtre numériquement les données stockées dans le vecteur \underline{e} avec le filtre décrit à la fois par le vecteur \underline{b} (coefficients du numérateur de $h(z)$) et le vecteur \underline{a} (coefficients du dénominateur de $h(z)$). Imposons au vecteur \underline{a} de commencer par un 1. Notons que la fonction « filter » permet tout autant de construire un filtre récursif (RII) qu'un filtre non récursif (RIF).
- ✓ « $[\underline{h}, \underline{\omega}] = \text{freqz}(\underline{b}, \underline{a}, N, f_e)$ » retourne N valeurs du gain complexe du filtre numérique échantillonné à la fréquence f_e (en Hertz), décrit par \underline{b} et \underline{a} . Ces valeurs sont stockées dans \underline{h} et calculées pour N pulsations stockées dans $\underline{\omega}$. Les pulsations sont equi-espacés sur l'intervalle $[0, \pi]$.
- ✓ « $[\underline{h}, \underline{t}] = \text{impz}(\underline{b}, \underline{a}, N, f_e)$ » retourne la réponse impulsionnelle du filtre numérique décrit par \underline{b} et \underline{a} . La réponse impulsionnelle est calculée en N instants stockés dans \underline{t} et espacés de $1/f_e$, les valeurs de réponse correspondante sont stockées dans \underline{h} .
- ✓ « $[\underline{b}, \underline{a}] = \text{butter}(n/2, \Delta f)$ » fournit les coefficients du filtre de Butterworth d'ordre n (pair de préférence). La bande passante du filtre doit être comprise entre 0 et $f_e/2$. Les bornes de cette bande passante, f_1 et f_2 , normalisées par $f_e/2$, sont stockées dans le vecteur Δf .
- ✓ « $[\underline{b}, \underline{a}] = \text{cheby1}(n/2, R_p, \Delta f)$ » fournit les coefficients du filtre de Chebyshev d'ordre n , de bande passante Δf et de R_p dB d'ondulations dans la bande passante.
- ✓ « $[\underline{b}, \underline{a}] = \text{ellip}(n/2, R_p, R_s, \Delta f)$ » fournit les coefficients du filtre de Cauer d'ordre n , de bande passante Δf , R_p dB d'ondulations dans la bande passante et R_s dB dans la bande atténuée.
- ✓ « $\text{stem}(\underline{t}, \underline{e})$ » donne une représentation graphique adéquate d'un signal numérique \underline{e} dont les instants d'échantillonnage sont stockés dans \underline{t} .
- ✓ « $\underline{s} = \text{fft}(\underline{e}, N)$ » calcule la transformée de Fourier discrète du signal numérique \underline{e} en N points à l'aide de l'algorithme Fast Fourier Transform (FFT). Afin d'accroître la rapidité de calcul, prendre pour n une puissance de deux.

1.3 Application

Comparer les gabarits des filtres passe bande de Butterworth, de Chebyshev et de Cauer d'ordre 2. On choisira pour chacun une fréquence d'échantillonnage de 800 Hz, une bande passante comprise entre 200 et 300 Hz. On

prendra également 6dB d'ondulation dans la bande passante et 20dB dans la bande d'atténuation. Puis augmenter l'ordre des filtres, que remarque-t-on ?

2 Estimation de gain complexe

2.1 Introduction

La meilleure manière de caractériser un filtre linéaire, continu, invariant (LCI) H est d'identifier sa réponse impulsionnelle h dans le domaine temporel ou bien son gain complexe dans le domaine fréquentiel $\text{TF}\{h\}$. De ce fait, nous allons ici proposer une méthode d'identification du gain complexe d'un système LCI à l'aide de la formule des interférences, supposant bien évidemment que pour toute entrée e il nous est possible de mesurer la sortie correspondante s du système.

2.2 Outils :

Hypothèses : Signaux supposés aléatoires stationnaires au sens large d'ordre 2 et ergodiques.

- ✓ Statistique d'ordre 1 (moyenne) : $m_e = E[e(t)]$
- ✓ Statistique d'ordre 2 (covariance) : $\gamma_{se}(\tau) = \text{Cov}(s(t), e(t-\tau)) = E[s(t) e(t-\tau)^*] - E[s(t)] E[e(t-\tau)]^*$
 $\gamma_e(\tau) = \text{Cov}(e(t), e(t-\tau)) = E[e(t) e(t-\tau)^*] - E[e(t)] E[e(t-\tau)]^*$
- ✓ Formule des interférences : $\text{TF}\{\gamma_{se}(\tau)\}(f) = \text{TF}\{\gamma_e(\tau)\}(f) \text{TF}\{h(t)\}(f)$
- ✓ Estimation empirique de $\gamma_{se}(\tau)$: $\hat{\gamma}_{se}(\tau) = N^{-1} \sum_{n=1}^{N-\tau} \{ (s(n+\tau) - \hat{m}_s) (e(n) - \hat{m}_e)^* \}$ pour $0 \leq \tau \leq N - 1$

2.3 Algorithme :

- ✓ Exciter le système avec un signal dont la densité spectrale soit aussi plate que possible sur toute la gamme de fréquences où la fonction de transfert est non nulle (générateur de bruit pseudo blanc).
- ✓ Estimer la densité spectrale $\text{TF}\{\gamma_{se}(\tau)\}(f)$ de l'intercorrélacion entre le signal d'entrée et le signal de sortie. La sortie est obtenue en filtrant le bruit pseudo blanc considéré comme signal d'entrée.
- ✓ Dédire une estimée du gain complexe du filtre considéré, et ce à partir de la formule des interférences en tenant compte de l'approximation $\hat{\text{TF}}\{\gamma_e(\tau)\}(f) \approx 1$ lorsque l'entrée est un pseudo bruit blanc.
- ✓ Réitérer K fois la manipulation et moyenner les K gains complexes obtenus : on dit ainsi moyenner sur K réalisations.

2.4 Simulation

- ✓ En pratique, on ne connaît ainsi pas la nature du système observé, on est juste en mesure de produire la sortie s correspondant à une entrée e donnée. Toutefois, nous allons ici nous placer dans la phase dite de « simulation », c'est-à-dire dans la phase de validation de l'algorithme : on choisit un filtre dont on connaît la réponse impulsionnelle ainsi que le gain complexe, on est alors en mesure de créer à partir d'une entrée, la sortie correspondante, de ce fait on va estimer le gain complexe du filtre en appliquant l'algorithme. Il nous sera alors possible de vérifier le résultat en le comparant au gain complexe exact et de tester l'efficacité de l'algorithme.
- ✓ Prendre pour filtre numérique le filtre de Butterworth d'ordre $n = 2$ avec pour bande passante celle employée au paragraphe précédent. Comparer graphiquement l'estimée du gain complexe avec l'original. Faites varier successivement K , N (le nombre d'échantillons) et n , qu'observez-vous ?
- ✓ Même exercice avec le filtre de Tchebychev d'ordre 2.