

### TD Prolongement sur les suites : les équations aux différences

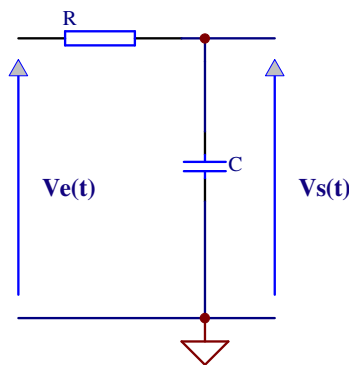
Nous pouvons généraliser l'idée d'exploiter des suites récurrentes d'ordre 1 ou 2 à un ordre  $n$ ... Ainsi, on peut même rendre récurrent les échantillons retardés de la sortie en plus de ceux de l'entrée : cette relation mathématique s'appelle l'équation aux différences.

Aussi, elle permet de passer à l'implémentation de filtres numériques à structure RIF (Réponse Impulsionnelle Finie) et RII (Réponse Impulsionnelle Infinie). Les filtres RIF ne considèrent que les échantillons retardés de l'entrée : on peut par exemple recréer l'écoute d'une salle de concert chez soi. Les filtres RII exploitent les échantillons retardés de la sortie et de l'entrée : ils permettent l'implémentation de filtre audio conventionnel ou vidéo.

Nous commencerons par écrire l'équation aux différences d'un filtre passe-bas du premier ordre.

#### *Vers l'équation aux différences*

##### Exercice 1 : Filtre passe-bas ramené à une équation aux différences



- 1) Ecrire l'équation différentielle du circuit ci-dessus.
- 2) Déterminer l'expression de la sortie  $V_s(t)$  pour une entrée  $V_e(t)$  en échelon d'amplitude  $E$ .
- 3) Substituer à l'équation différentielle une équation aux différences.
- 4) On donne :  $RC = 1$  ms,  $E = 1$  V et  $T_e = 0,2$  ms. Tracer  $V_s(t)$  et  $V_s(nT_e)$  en corrélation à l'aide de MATLAB.
- 5)  $T_e$  est maintenant égal à  $0,01$  ms. Tracer  $V_s(nT_e)$ . Conclure.
- 6) Utiliser cette équation aux différences par calculer la valeur moyenne d'un signal carré.

#### *Lien entre transformée en Z et équation aux différences*

##### Exercice 2 : Synthèse d'un oscillateur numérique

La transformée en  $z$  d'un signal sinusoïdal échantillonné  $x(nT_e)$  s'écrit :

$$H(z) = \frac{z^{-1} \sin \theta}{1 - 2z^{-1} \cos \theta + z^{-2}}$$

avec  $\theta = 2\pi f/f_e$ , pulsation normalisée.

En prenant  $f = 400$  Hz et  $f_e = 8$  KHz, passer à l'équation aux différences que l'on programmera avec Matlab afin d'obtenir le tracé de type sinusoïdal.

*Ecouter la musique comme si on était dans la salle...*

**Exercice 3 : Filtre RIF à phase non linéaire : salle de concert PORI**

Il s'agit d'effectuer un filtrage RIF d'une musique afin de reproduire l'écoute qu'un spectateur aurait dans une salle de concert. C'est le fameux effet concert hall que propose les ensembles « home cinema ».

Des mesures de réponses impulsionnelles ont été effectuées dans la salle de concert PORI (Finland) et disponibles sur leur site à l'adresse suivante :

<http://www.acoustics.hut.fi/projects/poririrs/>

1. Récupérer le fichier « poriref.pdf » qui explique le processus des mesures. Insister notamment sur les pages 4, 6 et 17 (omni.zip).
2. Récupérer le fichier « **s1\_r3\_o.wav** » :
  - a. Récupérer la fréquence d'échantillonnage et le nombre de bits utilisés pour coder l'échantillon (fonction audiinfo de Matlab).
  - b. Afficher la réponse impulsionnelle de chaque canal de cette mesure (fonction stem de Matlab). Remarques ?
  - c. Retrouver la position de la source S1 et la position du récepteur R3 dans la salle de concert PORI.
  - d. Tracer la représentation fréquentielle à l'aide de la fonction freqz. Remarques ?
3. Récupérer sur le réseau le fichier wav qui vous sera proposé lors de la séance de TD :
  - a. Récupérer la fréquence d'échantillonnage et le nombre de bits utilisés pour coder l'échantillon (fonction audioread de Matlab).
4. Récupérer sur le réseau le fichier « omni\_2023.zip » et dézipper-le dans votre répertoire de travail (Matlab) :
  - a. A l'aide du script « smarc.m », effectuer un « downsample » à 44100 Hz du fichier « **s1\_r3\_o.wav** ».
5. Effectuer le traitement numérique (en stéréo) à l'aide de la fonction filter.
6. Ecouter le résultat à l'aide de la fonction soundsc.
7. Enregistrer les échantillons traités dans un fichier wave à l'aide de la fonction audiowrite.