Physique

Capacité numérique 2 – Action d'un filtre sur un signal périodique

Pour le lundi 29 janvier 10:00

Pour rendre vos travaux (1 rendu par groupe de colle) :
— *les réponses aux questions « classiques » **par écrit** ;*
— *les codes à envoyer par mail à `l.torterotot.edu@gmail.com` au format*

`CN-Action_filtre-Gp_X.py`

*en remplaçant **X** par le numéro de votre groupe de colle.*

Objectifs

Simuler, à l'aide d'un langage de programmation, l'action d'un filtre sur un signal périodique dont le spectre est fourni. Mettre en évidence l'influence des caractéristiques du filtre sur le filtrage.

Afin d'utiliser les modules `numpy` et `matplotlib.pyplot`, importez-les dès le début de votre code avec :

```
import numpy as np
import matplotlib.pyplot as plt
```

1 Réponse en régime sinusoïdal d'un passe-bas d'ordre 1

Soit un filtre qui à un signal d'entrée $e(t)$ donne un signal de sortie $s(t)$. Ce filtre est décrit par une fonction de transfert $\underline{H}(\mathrm{j}\omega)$ telle que, en notation complexe,

$$\underline{S} = \underline{H}\,\underline{E}$$

avec $\underline{S}$ et $\underline{E}$ les amplitudes complexes respectives de s et e.

1. Dans le cas où $e(t) = E\cos(\omega t + \varphi_e)$, comment s'exprime $\underline{E}$ en fonction de E et φ_e ?

2. Écrire le code définissant la fonction `sinusoide`, prenant comme paramètres `A` l'amplitude, `omega` la pulsation, `phi` la phase à l'origine et `t` un `np.array` correspondant au temps en seconde et qui rend un `np.array` contenant les valeurs prises par $A\cos(\omega t + \varphi)$ aux instants t compris dans `t`.

3. Rappeler la forme canonique de la fonction de transfert $\underline{H}(\mathrm{j}\omega)$ d'un passe-bas d'ordre 1. Faire apparaître, en particulier, le gain en amplitude statique H_0 et la pulsation de coupure ω_0. Dans la suite, H_0 sera fixé à 1.

4. Écrire le code définissant la fonction de transfert générique du passe-bas d'ordre 1 `H_PasseBas1`, prenant comme paramètres `omega` la pulsation à laquelle évaluer $\underline{H}(\mathrm{j}\omega)$ et `omega0` la pulsation de coupure et qui rend un nombre complexe étant la valeur de $\underline{H}(\mathrm{j}\omega)$. Pour rappel, `1j` correspond en python à l'imaginaire pur j tel que $\mathrm{j}^2 = -1$.

5. Écrire le code définissant la fonction de transfert du filtre passe-bas d'ordre 1 ayant comme *fréquence* de coupure $f_{c1} = 500\,\mathrm{Hz}$, `Filtre_1`, prenant comme paramètres `omega` la pulsation à laquelle évaluer cette fonction de transfert un nombre complexe étant la valeur de cette fonction de transfert.

6. Dans le cas d'un filtre linéaire, le signal de sortie est de la forme $s(t) = S\cos(\omega t + \varphi_s)$. Comment s'exprime $\underline{S}$ en fonction de S et φ_s ?

7. Sachant que $\underline{S} = \underline{H}\,\underline{E}$, comment déterminer S et φ_s à l'aide de $\underline{H}$ et $\underline{E}$?

8. Écrire alors $s(t)$ à partir de E, ω, φ_e, $|\underline{H}|$ et $\arg(\underline{H})$.

9. Écrire le code définissant la fonction `sinusoide_filtree`, prenant comme paramètres `A` l'amplitude de la sinusoïde *non filtrée*, `omega` la pulsation, `phi` la phase à l'origine de la sinusoïde *non filtrée*, `t` un `np.array` correspondant au temps en seconde et `H` une fonction de transfert *elle-même fonction de* ω et qui rend un `np.array` contenant les valeurs prises par la sinusoïde en sortie du filtre. Utiliser dans `sinusoide_filtree` la fonction `sinusoide` précédemment définie.
Pour rappel, le module d'un nombre complexe `z` s'obtient avec `np.abs(z)`, son argument avec `np.angle(z)`.
Vous pourrez vérifier votre code avec les deux cas suivants :
- pour une fonction de transfert valant toujours 1, la sinusoïde filtrée est la même que la sinusoïde non filtrée $(s = e)$;
- pour une fonction de transfert valant toujours $\frac{1}{2}\mathrm{j}$, la sinusoïde filtrée correspond à celle en entrée mais d'amplitude divisée par 2 et en avance de $\pi/2$.

2 Série de Fourier pour un signal périodique

Un signal périodique n'est pas sinusoïdal. En revanche, le théorème de Fourier permet, en général, de décomposer un signal périodique en une somme de sinusoïdes.

10. Rappeler l'énoncé du théorème de Fourier pour un signal T-périodique.

Pour un signal C créneau d'amplitude E, c'est-à-dire

$$C(t) = \begin{cases} +E \text{ si } nT < t < \left(n+\frac{1}{2}\right)T \\ -E \text{ si } \left(n+\frac{1}{2}\right)T < t < (n+1)T \\ \quad 0 \text{ sinon} \end{cases} , \quad n \in \mathbb{Z},$$

il est possible de montrer que

$$C_n(t) = \sum_{k=0}^{n} c_k(t) = \sum_{k=0}^{n} \underbrace{\frac{4E}{(2k+1)\pi} \sin\left(2\pi(2k+1)\frac{t}{T}\right)}_{c_k(t)} \xrightarrow[n\to\infty]{} C(t)\,. \tag{1}$$

Dans la suite, nous fixerons la *fréquence* de ce créneau à 440 Hz et $E = 1$ SI.

11. Écrire le code de la fonction `Creneau`, prenant comme paramètres `E` l'amplitude du créneau $C(t)$, `T` la période du créneau $C(t)$, `t` un `np.array` correspondant au temps en seconde et qui rend un `np.array` contenant les valeurs prises par $C(t)$ en utilisant une boucle `for`. Il est possible de créer un `np.array` de longueur `L` et contenant uniquement des valeurs nulles grâce à `np.zeros(L)`.

12. Exprimer $c_k(t)$ à l'aide d'un cosinus et non d'un sinus. En déduire la fonction `C_n`, prenant comme paramètres `n` un entier, `E` l'amplitude du créneau $C(t)$, `T` la période du créneau $C(t)$, `t` un `np.array` correspondant au temps en seconde et qui rend un `np.array` contenant les valeurs prises par $C_n(t)$ en utilisant la fonction `sinusoide`.

13. Écrire un code permettant de tracer sur une même figure, sur 2 périodes, $C(t)$ le créneau exact et $C_n(t)$ pour $n = 0$ et de sauvegarder ce tracé dans une image « `fig1.png` ».

14. Faire de même pour $n = 1, 2, 500$ dans des images « `fig2.png` », « `fig3.png` », « `fig4.png` ».

3 Réponse du passe-bas d'ordre 1 à un signal périodique

Le filtre étant linéaire, « la sortie de la somme est la somme des sorties ».

15. Si s_k est la sortie du filtre pour la sinusoïde c_k (définie par (1)), exprimer la sortie du filtre S_n pour C_n en fonction de s_k.

16. En déduire le code de la fonction `S_n`, prenant comme paramètres `n` un entier, `E` l'amplitude du créneau $C(t)$, `T` la période du créneau $C(t)$, `t` un `np.array` correspondant au temps en seconde et `H` une fonction de transfert *elle-même fonction de* ω et qui rend un `np.array` contenant les valeurs prises par $S_n(t)$, la sortie du filtre de fonction de transfert $\underline{H}$ lorsque son entrée est $C_n(t)$. en utilisant la fonction `sinusoide_filtree`.

17.a. Écrire un code permettant de tracer sur une même figure la réponse temporelle, sur 2 périodes, $C(t)$ le créneau exact et la réponse du filtre `Filtre_1` à $C_{500}(t)$ et de sauvegarder ce tracé dans une image « `fig5-rep_temporelle.png` ».

b. Écrire un code permettant de tracer le spectre en amplitude de $C_{500}(t)$ et de la réponse du filtre à $C_{500}(t)$ entre 0 et 5 kHz et de sauvegarder ce tracé dans une image « `fig5-spectre.png` ». Pour cela, tracer pour chaque fréquence un trait allant de A à B avec $x_A = x_B = f$ la fréquence en Hz, $y_A = 0$ et y_B ayant la valeur de l'amplitude de la sinusoïde de fréquence f, pour chaque fréquence contenue dans la décomposition en série de Fourier du signal.

18. Faire de même avec une fréquence de coupure $f_{c2} = 5\,\mathrm{kHz}$ dans les images
« `fig6-rep_temporelle.png` » et« `fig6-spectre.png` ».

19. Faire de même avec une fréquence de coupure $f_{c3} = 50\,\mathrm{Hz}$ dans les images
« `fig7-rep_temporelle.png` » et« `fig7-spectre.png` ».

20. Commentez vos résultats, comparez les effets des différents filtres sur les réponses temporelles et fréquentielles.

4 Action d'un passe-bande d'ordre 2

21. Rappeler la forme canonique de la fonction de transfert $\underline{H}(\mathrm{j}\omega)$ d'un passe-bande d'ordre 2. Faire apparaître, en particulier, le facteur de qualité Q et la pulsation propre ω_0.

22. Écrire le code définissant la fonction de transfert du passe-bande d'ordre 2 `H_PasseBande2`, prenant comme paramètres `omega` la pulsation à laquelle évaluer $\underline{H}(\mathrm{j}\omega)$, `Q` le facteur de qualité et `omega0` la pulsation propre et qui rend un nombre complexe étant la valeur de $\underline{H}(\mathrm{j}\omega)$.

Les paramètres du filtre passe-bande seront, dans la suite, $Q = 1$ et $\omega_0 = 5 \times 2\pi/T$ avec T la période du créneau précédemment définie.

23. Déduire du travail effectué le code de la fonction `S_passe_bande_n`, prenant comme paramètres `n` un entier, `E` l'amplitude du créneau $C(t)$, `T` la période du créneau $C(t)$, `t` un `np.array` correspondant au temps en seconde et qui rend un `np.array` contenant les valeurs prises par la sortie du filtre passe-bande lorsqu'il est soumis à $C_n(t)$ en entrée.

24.a. Écrire un code permettant de tracer sur une même figure la réponse temporelle, sur 2 périodes, $C(t)$ le créneau exact et la réponse du filtre à $C_{500}(t)$ et de sauvegarder ce tracé dans une image « `fig8-rep_temporelle.png` ».

b. Écrire un code permettant de tracer le spectre en amplitude de $C_{500}(t)$ et de la réponse du filtre à $C_{500}(t)$ et de sauvegarder ce tracé dans une image « `fig8-spectre.png` ».

25. Commentez vos résultats.